

Grado en Ingeniería Electrónica Industrial y Automática
2018-2019

Trabajo Fin de Grado

“Automatización de una fuente de corriente-voltaje y un osciloscopio digital en el entorno de programación LabView”

Alejandro Moreno Martín

Tutor

Miguel Modesto Tardío

Leganés, 2019



[Incluir en el caso del interés de su publicación en el archivo abierto]

Esta obra se encuentra sujeta a la licencia Creative Commons

Reconocimiento – No Comercial – Sin Obra Derivada

RESUMEN

En este proyecto se va a desarrollar la automatización de dos equipos de laboratorio mediante la plataforma de programación LabVIEW. El proyecto se basará en la utilización de un Osciloscopio Yokogawa DL1540 y una SourceMeter Keithley 2410.

El objetivo es desarrollar una aplicación con una interfaz intuitiva y sencilla para el usuario y facilitar una tarea ardua como la toma de datos en un laboratorio. Se ha decidido utilizar las opciones mas comunes en cada dispositivo.

Se ha conseguido una aplicación funcional que permite al usuario manejar desde un ordenador los dos instrumentos y guardar los datos en un archivo externo.

En el desarrollo de este proyecto se va a describir una serie de conceptos teóricos sobre los instrumentos, las medidas y la plataforma de LabVIEW. Explicaremos como se ha ido desarrollando el trabajo y las dificultades con las que nos hemos encontrado.

ABSTRACT

In this project we will develop the automatization of two laboratory instruments with LabVIEW. The project will be based on the use of a Yokogawa DL1540 Oscilloscope and a Keithley 2410 SourceMeter.

The main objective is develop an application with a easy and friendly interface for the user facilitate an arduous task such as taking data in a laboratory. It has been decided to use the most common options in each device.

We have achived a funcional app that allows to the user manage since a computer both instruments and sabe data to a extern file

During this project, a series of theoretical concepts about the instruments, measurements and the LabVIEW platform will be described. We will explain how the work has been developed and the difficulties we have encountered

AGRADECIMIENTOS

A toda la gente que durante estos años me ha soportado.

A todos los que me llevo de esta etapa.

A mis padres por apoyar siempre incondicionalmente.

A mi abuelo, por darme cobijo durante estos años y llegar a ser de lo más importante de mi vida y a mi abuela por ser quienes somos gracias a ella.

INDICE

1. INTRODUCCIÓN Y OBJETIVOS	1
1.1. Antecedentes	2
2. FUNDAMENTOS TEÓRICOS DE LA INSTRUMENTACION	3
2.1. Las medidas eléctricas	3
2.2. Osciloscopio	3
2.2.1. Nuestros dispositivos	5
2.2.1.1. Osciloscopio Yokogawa DL1520	5
2.2.1.2. Fuente Keithley 2410	7
2.3. GPIB	9
2.3.1. SCPI	13
3. LA PLATAFORMA DE LABVIEW	15
3.1. ¿Qué es?	15
3.2. Panel frontal	15
3.3. Diagrama de bloques	15
3.4. Herramientas de programación	16
3.4.1. Paleta de herramientas	16
3.4.2. Paleta de funciones	17
3.4.3. Paleta de controles	18
3.5. Metodología de programación	18
3.6. Elementos más utilizados	20
3.6.1. While loop	20
3.6.2. Case structure	21
3.6.3. Sequence structure	21
3.7. Portabilidad de la aplicación	22
4. DESCRIPCIÓN DE LA SOLUCIÓN	24
4.1. La interfaz de la aplicación	24
4.1.1. Panel frontal	24
4.1.2. El osciloscopio	24
4.1.2.1. Configuración de parámetros pantalla	25
4.1.2.2. El trigger	26
4.1.3. La fuente	26
4.1.3.1. Modo rampa	27

4.1.3.2. Modo constante	28
4.2. La programación de la aplicación	29
4.2.1. La inicialización	29
4.2.2. El desarrollo.....	31
4.2.2.1. El osciloscopio.....	32
4.2.2.2. La fuente.....	41
4.3. Ejemplo de uso.....	47
4.4. Los datos	50
4.5. El ejecutable y el instalador	51
5. Conclusiones y futuras mejoras	55
6. Referencias	57
Anexo A: Presupuesto	1
Anexo B. Imágenes	3

ÍNDICE DE FIGURAS

Figura 2.1 Sistema de medida [1]	3
Figura 2.2 Funcionamiento osciloscopio [2]	4
Figura 2.3 Panel frontal osciloscopio [12]	5
Figura 2.4 Osciloscopio Yokogawa DL1520	6
Figura 2.5 SourceMeter Keithley 2410 [11].....	8
Figura 2.6 Modos de funcionamiento [11]	9
Figura 2.7 GPIB-USB-HS [14]	10
Figura 2.8 Detalle conector GPIB [5].....	11
Figura 2.9 Unión de varios conectores GPIB [6]	11
Figura 2.10 Pinout GPIB [13]	12
Figura 2.11 Detalle Pinout GPIB [13]	12
Figura 3.1 Ejemplo de subVI.....	16
Figura 3.2 Paleta de herramientas.....	16
Figura 3.3 Paleta de funciones.....	17
Figura 3.4 Paleta de controles.....	18
Figura 3.5 Ejemplo de subVI utilizado.....	19
Figura 3.6 Ejemplo de programación con comando SCPI en LabVIEW	19
Figura 3.7 Bucle "While Loop"	20
Figura 3.8 "Case structure"	21
Figura 3.9 Sequence structure	21
Figura 3.10 Creación de aplicación	22
Figura 3.11 Selección de VI a ejecutar.....	23
Figura 4.1 Pantalla principal.....	24
Figura 4.2 Detalle de la parte inferior.....	25
Figura 4.3 Menú de configuración de osciloscopio.....	25
Figura 4.4 Menú de configuración del trigger	26
Figura 4.5 Detalle de configuración de la fuente (Modo rampa, arriba, y modo constante, abajo)	27
Figura 4.6 Detalle de modos de rampa	27
Figura 4.7 Aspecto modo rampa	28
Figura 4.8 Detalle modo constante	28
Figura 4.9 Inicialización de la aplicación	29
Figura 4.10 Detalle de subVIs de inicialización.....	30
Figura 4.11 Secuencia de inicialización	30
Figura 4.12 Aspecto general del código	31
Figura 4.13 Botones de la aplicación.....	32
Figura 4.14 Configuración de aspectos de la interfaz	32
Figura 4.15 "Menu ring" del modo de funcionamiento de la fuente	33
Figura 4.16 Opciones de la fuente	33
Figura 4.17 "Menu ring" de modo osciloscopio o fuente con osciloscopio	33
Figura 4.18 Detalle del modo osciloscopio	34
Figura 4.19 "Menu ring" de la grafica de la fuente	34
Figura 4.20 Configuración del osciloscopio.....	34
Figura 4.21 Programación del osciloscopio	35
Figura 4.22 Programación "AutoSetup"	36

Figura 4.23 Comando SCPI para AutoSetup.....	36
Figura 4.24 Programación del trigger.....	37
Figura 4.25 Detalle de comando SCPI para configuración del trigger.....	37
Figura 4.26 Programación de "Run Oscilloscope"	38
Figura 4.27 Programación de la toma de datos del osciloscopio.....	39
Figura 4.28 SubVI de transformación de la gráfica.....	39
Figura 4.29 Detalle de "Save"	40
Figura 4.30 Programación de creación de "Paths"	40
Figura 4.31 Ejemplo de path creado	41
Figura 4.32 Creación array de datos para guardar	41
Figura 4.33 Detalle modo rampa de la fuente	42
Figura 4.34 Programación de constante	42
Figura 4.35 Ejemplo de configuraciones de pantalla posibles	43
Figura 4.36 Fuente en funcionamiento	44
Figura 4.37 "Menu ring" del modo rampa.....	44
Figura 4.38 Programación de modo rampa	44
Figura 4.39 Programación rectificación de puntos	45
Figura 4.40 Rectificación de los puntos	46
Figura 4.41 Programación modo normal	46
Figura 4.42 Estado inicial de la aplicación	47
Figura 4.43 Señal mostrada en la aplicación	48
Figura 4.44 Circuito diferenciador [3].....	48
Figura 4.45 Parámetros de modo rampa.....	49
Figura 4.46 Detalle de selección de modos de la grafica	49
Figura 4.47 Ejemplo de medidas	49
Figura 4.48 Datos tomados mostrados en una hoja de Excel	50
Figura 4.49 Gráfica creada con Excel	51
Figura 4.50 Proyecto de la aplicación	52
Figura 4.51 Selección de VI principal	53
Figura 4.52 Selección de aplicación a lanzar	53
Figura 4.53 Pantalla de selección de programas a instalar	54
Figura A.1 Fuente en funcionamiento	3
Figura A.2 Instrumentos y ordenador en marcha	3
Figura A.3 Detalle pantalla de osciloscopio acorde a figura 4.43	4
Figura A.4 Fuente externa para excitar señales cuadradas	4
Figura A.5 Aplicación corriendo en otro terminal	5

1. INTRODUCCIÓN Y OBJETIVOS

La numerosa cantidad de equipos de medición dentro de un laboratorio hace que la toma de medidas sea un trabajo largo si no se poseen los conocimientos y las técnicas necesarias para facilitar este trabajo. De esta problemática surge este proyecto, el simplificar y el ayudar al usuario a poder tomar datos con facilidad.

Mediante la automatización y la toma de datos a través de un elemento externo, en este caso un ordenador, vamos a poder simplificar y ayudarnos con las herramientas que posee a analizar todos los datos que obtengamos.

Para ello se ha optado por el empleo del software de National Instruments, LabVIEW. Una potente herramienta que permite la comunicación de los instrumentos con una aplicación que se usará para la toma de datos y la automatización del sistema.

La aplicación se comunicará a través del protocolo GPIB, mediante un adaptador a USB para el ordenador del mismo fabricante. Los equipos de instrumentación a automatizar han sido el osciloscopio Yokogawa 1520L y la fuente Keithley 2410. El conjunto de estos dos equipos permite la entrada de una serie de señales mediante la fuente y la salida de datos mediante el osciloscopio.

Los objetivos han sido los siguientes:

- Simplificar y disminuir el tiempo de configuración de los equipos para su funcionamiento.
- Entender los modos de funcionamiento más complejos de los equipos de medida.
- Acercar al usuario una interfaz gráfica más intuitiva y cómoda mediante las herramientas que proporciona LabVIEW

El desarrollo de este proyecto ha sido a través de LabVIEW 2017 con una licencia de estudiante proporcionada a través de un acuerdo con la universidad.

El proyecto está estructurado para poder usar solo la fuente, solo el osciloscopio o ambos al mismo tiempo.

1.1. Antecedentes

En la actualidad prácticamente todos los fabricantes de equipos electrónicos de medida incorporan diferentes puertos de comunicación para conectarlos a otros dispositivos. Existen conexiones RS232, GPIB o más modernas, USB. Nosotros vamos a decantarnos en este proyecto por la conexión GPIB, que explicaremos más adelante.

Elegimos GPIB ya que uno de los instrumentos tiene más de 20 años de antigüedad por lo que carece de USB. Usaremos eso sí, un adaptador de GPIB a USB para poder comunicarnos con un ordenador portátil. También serán de utilidad los drivers y subVIs que proporciona el fabricante de cada equipo.

Ampliando el último concepto, hay que tener en cuenta, ya que son parte muy importante del desarrollo del proyecto, estos subVIs. Son pequeñas partes de código de LabVIEW que ejecutan diferentes funciones específicas.

Aquí es donde viene el problema a resolver. Todas por si solas no tienen ninguna utilidad, ya que nadie quiere que un osciloscopio solo haga un autoajuste, o que una fuente cambie la configuración mostrada en la pantalla.

Para conseguir un programa útil, es necesaria la combinación de muchas de ellas, y ese va a ser el objetivo del proyecto. Conseguir la unión de diferentes minifunciones (subVI) y crear una gran aplicación que aúna todo.

2. FUNDAMENTOS TEÓRICOS DE LA INSTRUMENTACION

2.1. Las medidas eléctricas

Lo primero que se debe explicar es lo que es una medición eléctrica. Las mediciones eléctricas son cualquier acción que necesita dispositivos, cálculos o métodos para cuantificar una medida eléctrica. Estas señales son creadas al usarse transductores o propiedades físicas como fuerzas, temperaturas o presión. Estas propiedades son variables por lo que se utilizan sensores para poder transformar esa propiedad en una señal eléctrica.

Un sistema completo de medida está compuesto por varios elementos.

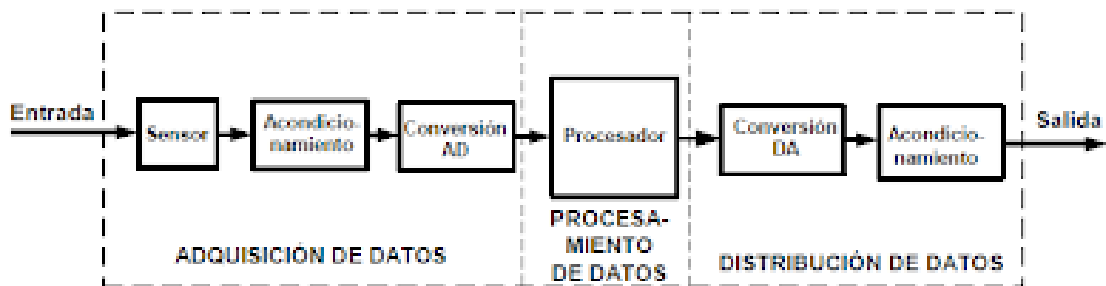


Figura 2.1 Sistema de medida [1]

Estos elementos son necesarios en cada medida que se haga o realice. En primer lugar, se necesita un elemento encargado de adquirir la medida, es decir, sensores que tomen los datos. A continuación, se debe hacer un acondicionamiento de la señal, que transforme esa señal en algo más legible. Una vez transformado, se hace una conversión A/D y se procesan los datos.

Cabe destacar que, en caso necesario de tener la medida en un formato analógico, se realiza una nueva conversión, en este caso D/A, y se vuelve a acondicionar la señal.

Debido a la finalidad del proyecto, se va a detallar más profundamente el funcionamiento de los osciloscopios, un elemento que combina todos los pasos anteriores

2.2. Osciloscopio

La gente que se maneja en este mundo de la ingeniería puede llegar a conocer y manejar un osciloscopio, pero este elemento es uno de los más importantes dentro del mundo de la electrónica. Es un instrumento de medición que posee una pantalla que muestra la

amplitud en vertical y el tiempo en la horizontal. Normalmente se necesita otro instrumento para poder tener una lectura en el osciloscopio.

Estos instrumentos muestran la información en dos ejes, formando el llamado oscilograma. Estos dispositivos disponen de diversos controles en su frontal para la configuración de la información en la pantalla.

Existen dos tipos de osciloscopios, los digitales y los analógicos. Los analógicos utilizan una serie de elementos ya en decadencia y que los hace menos precisos.

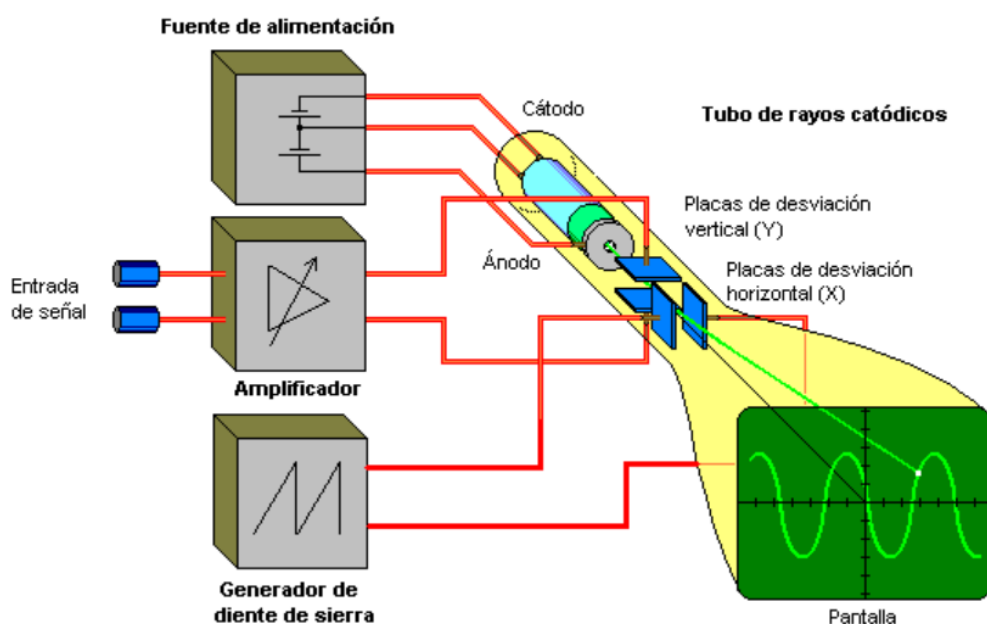


Figura 2.2 Funcionamiento osciloscopio [2]

En la figura se puede ver una representación del analógico. La tensión se aplica sobre las placas verticales y sobre las horizontales se aplica una tensión en diente de sierra producida mediante un oscilador y con frecuencia regulable. El tubo de rayos catódicos proporciona un rayo que, generado por el cátodo y acelerado por el ánodo, llega a la pantalla que se ilumina por el impacto de los electrones.

Estos tipos de osciloscopios tienen numerosas desventajas. Para empezar, han de ser periódicos, ya que esa periodicidad es la que hace que se refresque la pantalla. También las señales muy rápidas pueden reducir el brillo de la pantalla, ya que la tasa de refresco disminuye, además, solo se pueden observar los transitorios si estos son repetitivos.

En cuanto a los osciloscopios digitales, estos están reemplazando en gran medida a los analógicos. En estos osciloscopios, la señal es digitalizada por conversores. Poco más se puede añadir al funcionamiento de estos, ya que todo son circuitos electrónicos y la parte

analógica desaparece. Este tipo de osciloscopio añade funciones extra que con los analógicos sería imposible de realizar. Son capaces de medir automáticamente valores de pico, máximos y mínimos, capturas de transitorios, o incluso realizar capturas de pantallas para almacenar en dispositivos externos.

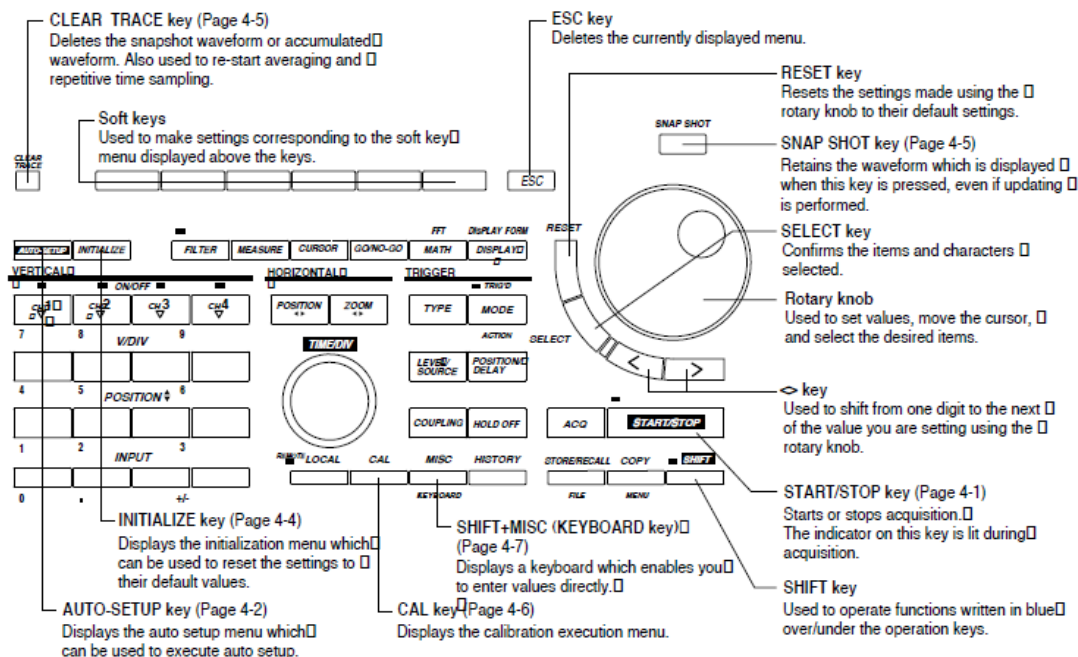


Figura 2.3 Panel frontal osciloscopio [12]

En esta imagen se observa el esquema del panel frontal. Todos los botones tienen su función, pero se va a explicar los más importantes durante el desarrollo, ya que sería una ardua tarea explicar el funcionamiento completo.

2.2.1. Nuestros dispositivos

Los dispositivos que vamos a utilizar son el osciloscopio Yokogawa DL1520 y la fuente de alimentación Keithley 2410.

2.2.1.1. Osciloscopio Yokogawa DL1520

Como se ve en la foto, este dispositivo tiene una cantidad de años importante, exactamente está fabricado entorno a mediados de los años noventa. Es un osciloscopio

digital, aunque su aspecto y su pantalla de lugar a confusiones. Como hemos dicho anteriormente en las características de los osciloscopios digitales, este dispositivo es capaz de realizar todas las funciones que realiza este modelo.



Figura 2.4 Osciloscopio Yokogawa DL1520

Como se ve en la imagen, este modelo tiene una serie de botones y funciones que vamos a detallar, ya que estos mismos comandos se van a realizar desde la aplicación.

El osciloscopio presenta dos canales. Ambos con el conector habitual de los osciloscopios, el BNC. Estos dos canales son totalmente independientes y cada uno se puede configurar de la manera que se desee. El problema que tiene este osciloscopio es que pese a ser digital, la pantalla es monocromática, todo lo que aparece en la pantalla tiene un color anaranjado.

Para poder configurar cada canal se tienen diferentes opciones, describiremos las más importantes. La primera y de las más importantes es la escala de amplitud de los canales. Cada canal se puede configurar en este apartado independientemente, haciendo que cada canal tenga su propia escala y diferenciando señales con amplitudes bajas y altas. Para

ello, seleccionando la opción de V/div del canal que se quiere modificar, y rotando la ruleta derecha, se configura al gusto.

Otro parámetro importante por configurar es la escala de tiempo de la pantalla. Este parámetro es común a ambos canales y para ello ajustando la ruleta que se localiza en el centro del dispositivo, se ajusta cuantos segundos o incluso nanosegundos, queremos que sea cada división en la pantalla.

También otra opción que presentan estos osciloscopios es la posibilidad de mover, tanto horizontalmente como verticalmente, por la pantalla los datos mostrados. Para ello existe la opción de “position”. Si bien es cierto que la posición vertical si es ajustable en cada canal, la posición horizontal es común, ya que la escala de tiempo es la misma. Para esta opción existen tres botones, dos situados a la izquierda para el movimiento vertical, y otro situado en el centro para el horizontal. Todos, al pulsarlos, activa esta opción, que luego es configurable con la ruleta derecha.

Todo esto es una configuración totalmente manual, que al fin y al cabo la configuramos a nuestras necesidades. Nosotros somos los que mejor conocemos el sistema a medir.

2.2.1.2.Fuente Keithley 2410

El segundo instrumento que vamos a usar una fuente de alimentación algo especial, tanto que hasta el nombre es diferente. Este instrumento se le conoce como SMU (SourceMeter Unit). Este dispositivo es capaz de excitar con su fuente de tensión o corriente y medir lo que se desee a la salida de esta, de manera que hace las veces de fuente y multímetro.

Esta unidad tiene la capacidad de proporcionarnos una corriente de entre 1.05A y 10pA y una tensión de entre 1100 V y 1μV con una potencia de 20 W.



Figura 2.5 SourceMeter Keithley 2410 [11]

En la imagen superior vemos el panel frontal de Keithley 2410. Se diferencian dos zonas de funcionamiento, fuente (source) y medida (measure). Por el lado de las medidas se observan los tres parámetros que se pueden medir, tensión, intensidad y resistencia. Y por el lado de la fuente, la tensión y la corriente. Al seleccionar cada una de estas opciones no da la opción de configuración de cada modo. Las medidas siempre se muestran en la parte superior de la pantalla, en un tamaño mayor al de la fuente, que se muestra en la parte inferior.

Junto con los botones de configuración de medidas y fuente, se observan las diferentes opciones que posee este instrumento. Cabe destacar entre otros, el modo sweep (barrido), un modo que permite configurar una señal en tiempo y valor, pudiendo hacer desde rampas a señales totalmente configurables.

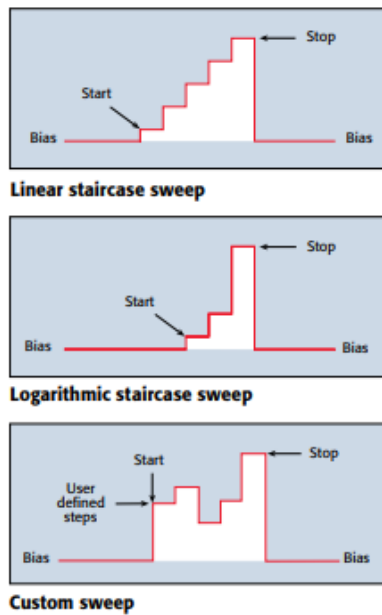


Figura 2.6 Modos de funcionamiento [11]

Otras opciones que pose son el trigger de la fuente, al igual que muchos otros instrumentos, el límite de corriente o diferentes filtros que se pueden configurar. También un botón importante para nosotros es el botón “Local”, que permite retomar el control del panel frontal de la SourceMeter cuando la aplicación está corriendo en el ordenador.

Este instrumento como prácticamente todos los que generan una señal, tiene la opción de limitar la intensidad de salida para proteger el sistema a excitar. Esto se hace con la “Compliance”. Este valor limita tanto la corriente como la tensión de salida para proteger le sistema de medida.

2.3. GPIB

Para la conexión entre el ordenador y los instrumentos se utiliza el bus de comunicación GPIB. Este puerto, presente en los instrumentos de medida de un laboratorio permite la comunicación entre estos y el ordenador principal. GPIB permite la conexión en paralelo de gran cantidad de dispositivos, lo que hace convertirse en la mejor opción para nuestro trabajo.

El protocolo de comunicación GPIB fue desarrollado durante los años 60 por HP, donde al principio lo denominaban HPIB. Es un protocolo dedicado a conectar y controlar instrumentos de laboratorio, como fuentes, osciloscopios o demás instrumentación. En

1975 el Instituto de Ingenieros Eléctricos y Electrónicos (IEEE) lo bautizo como IEEE 488 recogiendo la especificaciones eléctricas, mecánicas y funcionales de la interfaz. Es por eso por lo que se puede conocer como GPIB, HPIB o IEEE 488.

La primera especificación de GPIB traía algunos problemas por lo que se lanzó una nueva especificación el IEEE 488.2, que incluye códigos, formatos, protocolos y comandos comunes. En 1990 el bus incluyó los Comando Estándares para la instrumentación Programable (SCPI), por lo que se definían los comandos que tenían que cumplir cada clase de instrumento, por lo que así se aseguraba el correcto funcionamiento entre diferentes instrumentos.

Para nuestro proyecto se utiliza un adaptador GPIB a USB, ya que los ordenadores más actuales del mercado no incorporan el puerto GPIB de serie. Este dispositivo es fabricado por National Instruments y permite la conexión de un instrumento con un ordenador.



Figura 2.7 GPIB-USB-HS [14]

Debido a que este adaptador solo permite la comunicación de un único instrumento, es necesario tener otro cable para poder conectar dos dispositivos.



Figura 2.8 Detalle conector GPIB [5]

Estos cables permiten ir encadenando dispositivos según sean necesarios, conectando como se ve en la imagen siguiente.



Figura 2.9 Unión de varios conectores GPIB [6]

Con la combinación de varias conexiones de este modo, se permite llegar hasta un máximo de 15 dispositivos en paralelo.

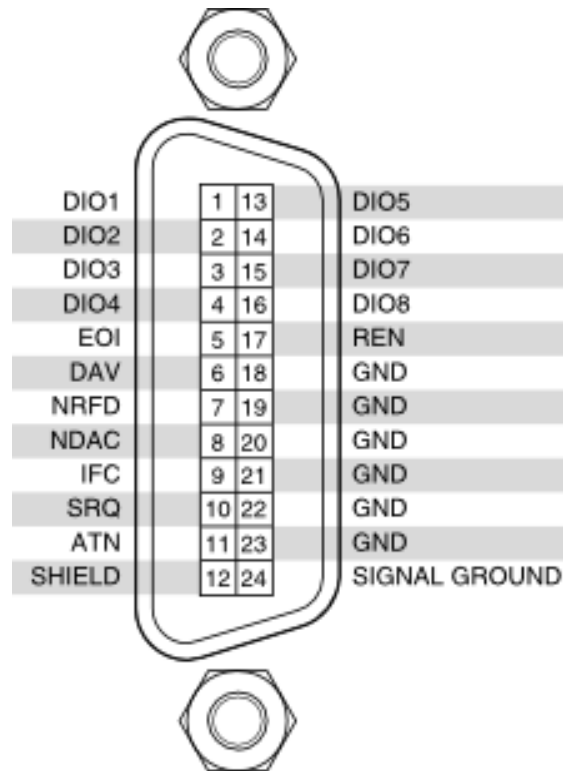


Figura 2.10 Pinout GPIB [13]

La conexión GPIB posee el siguiente pinout en los conectores, cada uno con su señal correspondiente.

Signal	Terminal	Description
DIO1	1	Data Input/Output Bit.
DIO2	2	Data Input/Output Bit.
DIO3	3	Data Input/Output Bit.
DIO4	4	Data Input/Output Bit.
EOI	5	End-Or-Identify.
DAV	6	Data Valid.
NRFD	7	Not Ready For Data.
NDAC	8	Not Data Accepted.
IFC	9	Interface Clear.
SRQ	10	Service Request.
ATN	11	Attention.
SHIELD	12	Shield.
DIO5	13	Data Input/Output Bit.
DIO6	14	Data Input/Output Bit.
DIO7	15	Data Input/Output Bit.
DIO8	16	Data Input/Output Bit.
REN	17	Remote Enable.
GND	18	Ground—Wire twisted with DAV.
GND	19	Ground—Wire twisted with NRFD.
GND	20	Ground—Wire twisted with NDAC.
GND	21	Ground—Wire twisted with IFC.
GND	22	Ground—Wire twisted with SRQ.
GND	23	Ground—Wire twisted with ATN.
SIGNAL GROUND	24	Logic Ground.

Figura 2.11 Detalle Pinout GPIB [13]

Este protocolo posee grandes ventajas ya comentadas anteriormente, aunque también permite la conexión de dispositivos de diferentes fabricantes y características técnicas o como que la conexiones son muy robustas entre los diferentes equipos. Pero no todo son ventajas, este sistema por lo general es mucho más caro, ya que se necesitan cables específicos para cada uno de los dispositivos, e incluso el montaje suele ser más lento en una instalación así. Otra de las desventajas que tiene es la falta de estándares para la programación, ya que depende de cada fabricante los comandos de programación que se usan para cada dispositivo, haciendo la programación a bajo nivel muy compleja, teniendo el desarrollador que aprender para cada fabricante y dispositivo, como usarlo.

El problema de los estándares se palió un poco con el llamado SCPI.

2.3.1. SCPI

SCPI (Standard Commands for Programmable Instruments) es un estándar para la programación de instrumentos de laboratorio basado en comandos de palabras para comunicar un dispositivo con otro. El principio de funcionamiento es simple, ya que los comandos son textos con caracteres ASCII.

Existen dos tipos de comandos:

- Comandos comunes: Son aquellos que aparecen y son iguales en todos los instrumentos. Son los dedicados a resetear (RESet) o identificación de instrumentos, (IDN?)
- Comandos de control: son los que permiten la comunicación con un instrumento y presentan una estructura jerarquizada. Dentro de los comandos de control existen de dos tipos:
 - Operaciones de consulta, que son los encargados de mostrar la información que el instrumento nos está dando. Estos comandos terminan con una interrogación “?”.
 - Operaciones de ajustes, que nos dan la posibilidad de poder configurar nuestro instrumento dando las ordenes pertinentes.

La estructura de los mensajes SCPI es la siguiente. Estos mensajes están organizados en una estructura de árbol, que nos permite navegar de arriba abajo para llegar a nuestro destino.

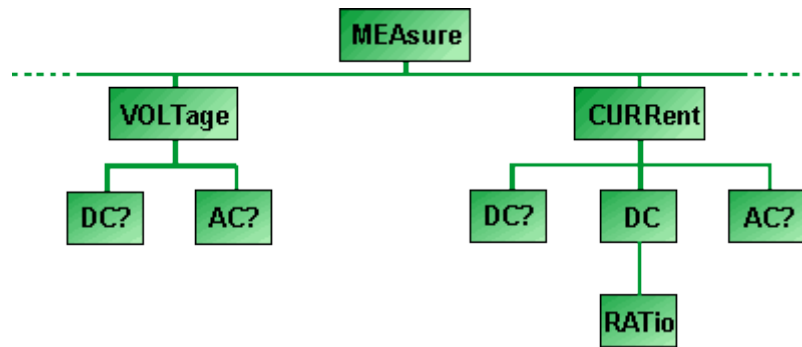


Figura 2.12 Ejemplo de composición comando SCPI [8]

Como se ve en la imagen anterior, este es un ejemplo del comando **MEASURE**. Ese comando a su vez puede ser de **VOLTage** y **CURRent** y a su vez tienen más submensajes. Para poder mandar la orden de querer medir el voltaje en corriente continua que el instrumento está midiendo en ese momento la estructura es la siguiente:

MEASURE:VOLTage:DC?

Cabe destacar la estructura de escritura, donde cada vez que se baje un nivel es necesario escribir los dos puntos ":", al igual que no es obligatorio incluir en los comandos los caracteres escritos en minúscula, con escribir **MEA:VOLT:DC?**, sería suficiente.

3. LA PLATAFORMA DE LABVIEW

3.1. ¿Qué es?

LabVIEW es una plataforma de desarrollo de aplicaciones que requieran medidas y control sobre un proceso o aplicación industrial o de ingeniería desarrollada por National Instruments (NI). Permite la programación de instrumentos virtuales a través de un lenguaje gráfico de bloques y operaciones. Para este proyecto se ha utilizado esta plataforma debido a las facilidades que ofrece para la utilización del protocolo GPIB. National Instruments proporciona una serie de drivers que agilizan su integración.

El principio fundamental de LabVIEW es la utilización del mencionado lenguaje gráfico. Este se basa en dos partes, los llamados diagramas de bloques y el panel frontal. El primero es el encargado de albergar la programación. En él se colocan los diferentes bloques de programación. Estos bloques pueden ser funciones básicas de programación, estructuras y los llamados VI (Virtual Instrument), muy importantes en este proyecto y que más adelante veremos en detalle. La otra parte, el panel frontal es la interfaz gráfica de usuario. En ella se verá de una forma cómoda los menús y botones que servirán para controlar la aplicación. LabVIEW se comunica con los instrumentos mediante el protocolo GPIB.

3.2. Panel frontal

Es la interfaz gráfica de usuario. En ella se sitúan los controles y los indicadores de la aplicación para la interacción con el usuario. Para ello se tiene diferentes opciones de personalización de los controles e indicadores. También pueden existir gráficas, potenciómetros, reguladores y demás controles para poder tener una experiencia completa.

3.3. Diagrama de bloques

Es quizás la parte más importante de una aplicación de LabVIEW. Es una parte oculta para el usuario final pero la que contiene el código fuente que permite el funcionamiento de la aplicación. Como se ha dicho anteriormente, LabVIEW utiliza una programación gráfica utilizando una serie de herramientas como pueden ser bloques booleanos, arrays, cadenas de caracteres, e incluso los llamados VI. Todo se conecta entre si mediante, diversos métodos como cables, variables locales o los llamados nodos de propiedad.

Las aplicaciones suelen tener un VI principal sobre el que se van colocando los distintos subVI y elementos de programación. Para ello la estructura de un VI tiene que ser coherente, por lo que poseen una serie de entradas y salidas para poder relacionarlos entre si.

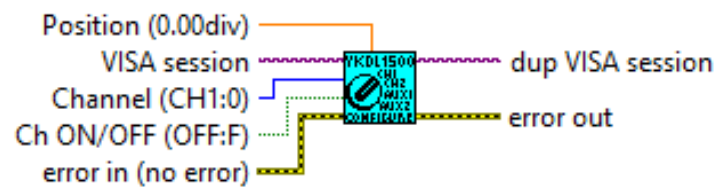


Figura 3.1 Ejemplo de subVI

Como se ve en la imagen, los elementos de la izquierda son las llamadas entradas, que a su vez pueden ser constantes o controles y los elementos de la derecha son las salidas, que se pueden monitorizar a través de los indicadores.

LabVIEW utiliza un código de figuras y bloques que dependiendo de la utilidad y la finalidad se combinan para formar el código del programa. También se utiliza un color de cable y tamaño diferente dependiendo de los datos que transportan

3.4. Herramientas de programación

Para la construcción de los códigos existen una serie de herramientas que permiten la interacción con el diagrama de bloques.

3.4.1. Paleta de herramientas

La primera de ellas es la paleta de funciones, la encargada de poder elegir el tipo de acción que se desea realizar. Esta paleta tiene dos modos de funcionamiento, el automático y el manual.



Figura 3.2 Paleta de herramientas

El automático se activa con el control superior de la paleta. En este modo la herramienta se selecciona automáticamente dependiendo de la zona sobre la que se sitúe. El modo manual permite seleccionar en cada momento la herramienta que se desee, evitando los errores de uso que puede provocar el modo automático.

La paleta tiene diferentes herramientas: crear cables de conexión, seleccionar elementos del diagrama de bloques o panel frontal, poder colorear elementos, o poder escribir textos.

3.4.2. Paleta de funciones

Únicamente se encuentra disponible en el diagrama de bloques. En ella se encuentra disponible toda una serie de funciones que permiten la programación de la aplicación. Hay desde funciones matemáticas, (sumas, multiplicaciones...), funciones booleanas y temporizadores hasta herramientas más complejas como estructuras o tratamientos de datos.

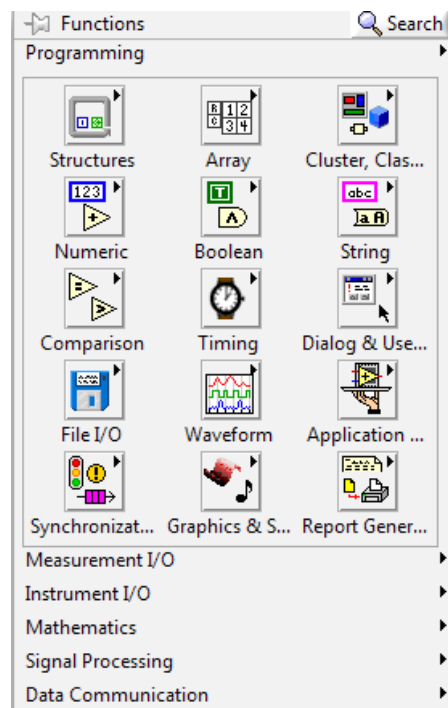


Figura 3.3 Paleta de funciones

La unión de los diferentes elementos disponibles en la paleta va formando el programa deseado. Algunas de las subpaletas son subVI proporcionados por el desarrollador que realizan diferentes funciones.

3.4.3. Paleta de controles

Esta paleta solo está disponible en el panel frontal. Son los elementos que proporcionan la información visual de la aplicación.

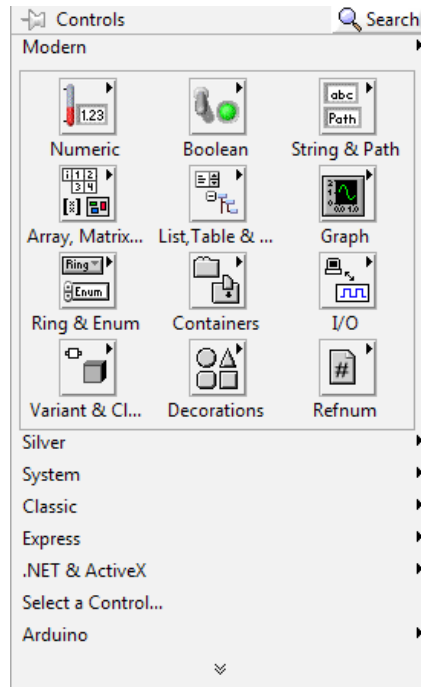


Figura 3.4 Paleta de controles

Esta paleta posee todos los elementos que son visibles para el usuario final. Desde indicadores numéricos, booleanos e indicadores textuales, hasta gráficas, e incluso elementos decorativos. Cada vez que uno de estos elementos es colocado en el panel frontal, en el diagrama de bloques se crea el mismo elemento para su tratamiento en el código del VI.

3.5. Metodología de programación

LabView, como ya se ha comentado, es un lenguaje de programación gráfico. Esto quiere decir que toda la programación se desarrolla en el llamado diagrama de bloques. En este diagrama se utilizan todas las herramientas mostradas en la paleta del diagrama de bloques, pero no solo se utilizan esas herramientas. Dependiendo del tipo de aplicación que queremos hacer, el fabricante del hardware da unos elementos llamados drivers o controladores. Estos drivers son lo que se llaman subVI, que no dejan de ser pequeñas aplicaciones desarrolladas por el fabricante para ayudar a la programación modular de la aplicación.

La programación con LabView sigue las mismas reglas y métodos que cualquier otro lenguaje, debe estar siempre bien estructurado, siguiendo un flujo de datos que permite que la aplicación siga un desarrollo lógico. Para programar con LabView es importante tener en cuenta que lo que en cualquier lenguaje serian funciones, aquí son los anteriormente llamados drivers.

La unión de elementos de programación de LabView junto con los drivers proporcionados por diversos desarrolladores, crea una aplicación modular. Los drivers contienen en su interior una aplicación que realiza una determinada tarea, que a nivel práctico para el desarrollador no afecta lo que se encuentre en su interior, solo lo que es capaz de realizar.

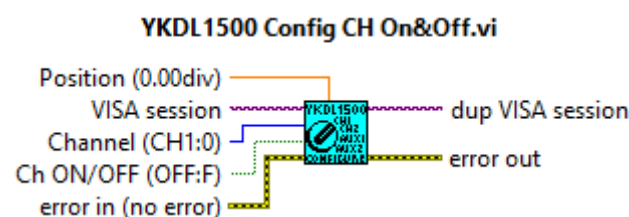


Figura 3.5 Ejemplo de subVI utilizado

Como se ve en la imagen, así luce un driver o controlador proporcionado en este caso por el fabricante del osciloscopio. Internamente no importa que es lo que hay dentro de él, de hecho si lo abrimos y vemos su código luce así.

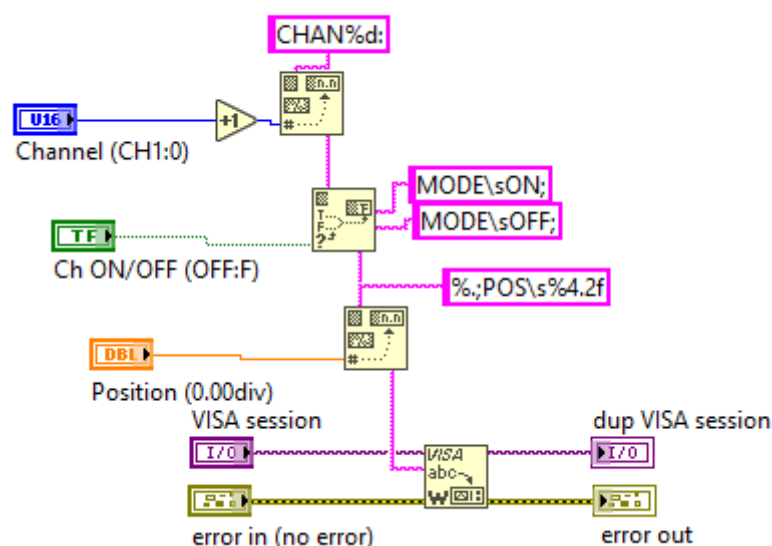


Figura 3.6 Ejemplo de programación con comando SCPI en LabVIEW

En este caso no es uno de los drivers más complejos que nos podemos encontrar, pero a la hora de realizar la función para la que está hecho, no es obligatorio entender que realiza internamente.

Aprovechando la imagen anterior, vemos que aparecen algunos de los elementos mas comunes en LabView y que detallaremos mas adelante. En este caso aparecen los llamados controles, que son valores que el usuario puede modificar. De arriba abajo encontramos, de color azul un control numérico de 16 bits, en verde un control booleano, en naranja un control numérico de tipo doble, 32 bits con signo, en morado, el control de la VISA sesión (Virtual Instrument Software Architecture) y en amarillo el control de los mensajes de errores. También se ven en rosa las contantes de strings, así como un elemento de suma, que añade al control de 16 bits n uno.

3.6.Elementos más utilizados

Dentro de la paleta de funciones del diagrama de bloques se encuentran las diferentes herramientas para la programación. Algunas de la más utilizadas son las siguientes.

3.6.1. While loop

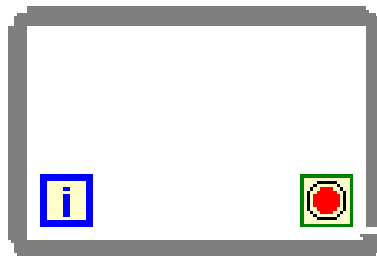


Figura 3.7 Bucle "While Loop"

Este elemento es uno de los más utilizados, permite la ejecución en bucle de lo que contenga en su interior. Para su buen uso hay que tener en cuenta una serie de advertencias. Siempre es recomendable colocar en su interior un elemento temporal, para así poder ejecutar el bucle a una cierta frecuencia, ya que, de lo contrario, el bucle se ejecutaría a la máxima velocidad posible del procesador del equipo que se esté utilizando, lo que provoca un consumo de energía y recursos altos.

En este tipo de bucles, es obligatorio cablear el botón de abortar. En este botón debe ir conectado un control booleano, que dependiendo de la funcionalidad que queramos, puede

ser un botón manual, un cable de alguna condición que se realice interior o exteriormente o una variable que se haya creado anteriormente.

3.6.2. Case structure

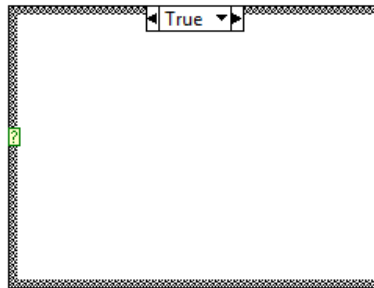


Figura 3.8 "Case structure"

Esta estructura es otra de las más utilizadas en LabView. Su función es la de poder seleccionar que secuencia o parte de código que se realiza cuando se cumple la condición del selector. La más común es el selector de TRUE o FALSE, aunque también puede ser un selector numérico o de tipo string. Cuando el selector recibe un dato, dependiendo del valor de este, se realiza el subdiagrama que se encuentra en ese caso.

3.6.3. Sequence structure

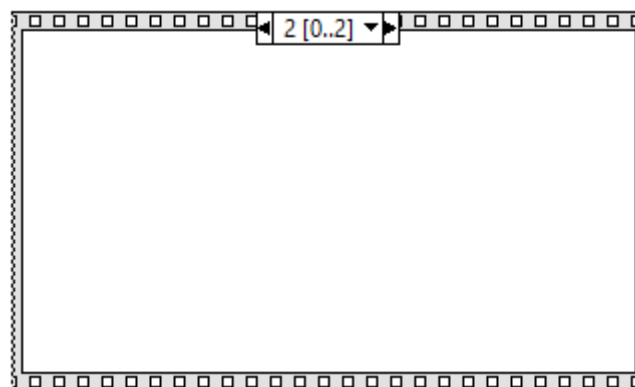


Figura 3.9 Sequence structure

Otra de las estructuras más utilizadas son las secuencias. Estas estructuras permiten la secuenciación de un programa. El código presente en el interior de esta estructura se irá ejecutando secuencialmente y hasta que no se termine un subdiagrama, no se pasará al siguiente. Suele ser útil en el momento de la inicialización, ya que antes de comentar con

la ejecución principal, se puede programar una secuencia que permita la configuración inicial de la aplicación.

3.7. Portabilidad de la aplicación

LabView, dentro de sus opciones, permite la creación de aplicaciones portables para su uso en otros sistemas. Estas aplicaciones son creadas en formato .exe, lo que permite su uso en cualquier sistema que tenga instalado previamente LabView, sin necesidad de tener el código fuente en ningún sitio.

Para la creación de estas aplicaciones, se utiliza una herramienta presente en la pantalla principal del proyecto.

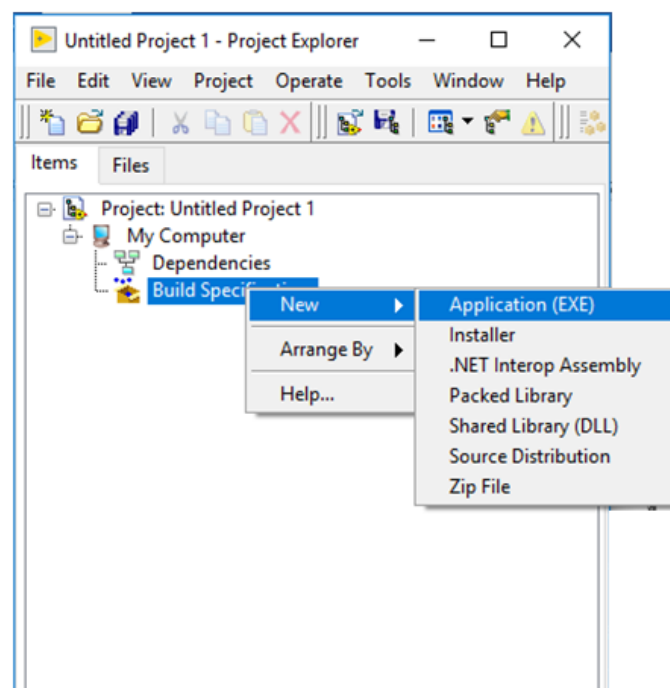


Figura 3.10 Creación de aplicación

Esta herramienta abre una ventana que permite la creación del .exe.

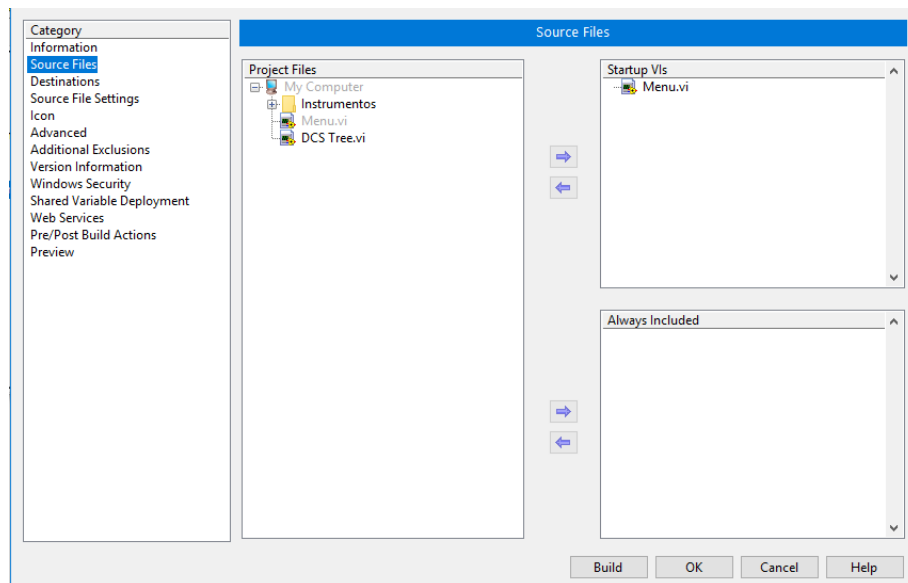


Figura 3.11 Selección de VI a ejecutar

En esta ventana, hay que seleccionar el VI que queremos que se ejecute en modo principal, es decir, el VI que será el más importante y el que desencadena el uso de otros subVI's.

En este caso, la aplicación .exe se ha podido ejecutar fácilmente en cualquier ordenador del laboratorio, pero existe otra opción de poder ejecutar nuestra aplicación en cualquier parte.

LabView permite la opción de crear un archivo instalador para poder usar la aplicación en otro dispositivo que no tenga instalado ninguna versión de LabView. Este archivo se crea de la misma forma que los ejecutables .exe. Al seleccionar la opción installer, se abre una ventana similar a la de los ejecutables. La única diferencia que este paso lleva mas proceso detrás por parte de LabView. Para poder crear este instalador, se añaden al .exe que se crea, una serie de archivos instalables extra que, al ejecutarlos en otro ordenador, instala en el nuevo dispositivo los programas de National Instruments necesarias para la ejecución

4. DESCRIPCIÓN DE LA SOLUCIÓN

4.1. La interfaz de la aplicación

Con lo comentado en el apartado tres ya podemos pasar a describir cómo se va a estructurar la aplicación.

4.1.1. Panel frontal

Lo primero de todos es explicar cómo es la interfaz de usuario, es decir, el panel que manejará el usuario.

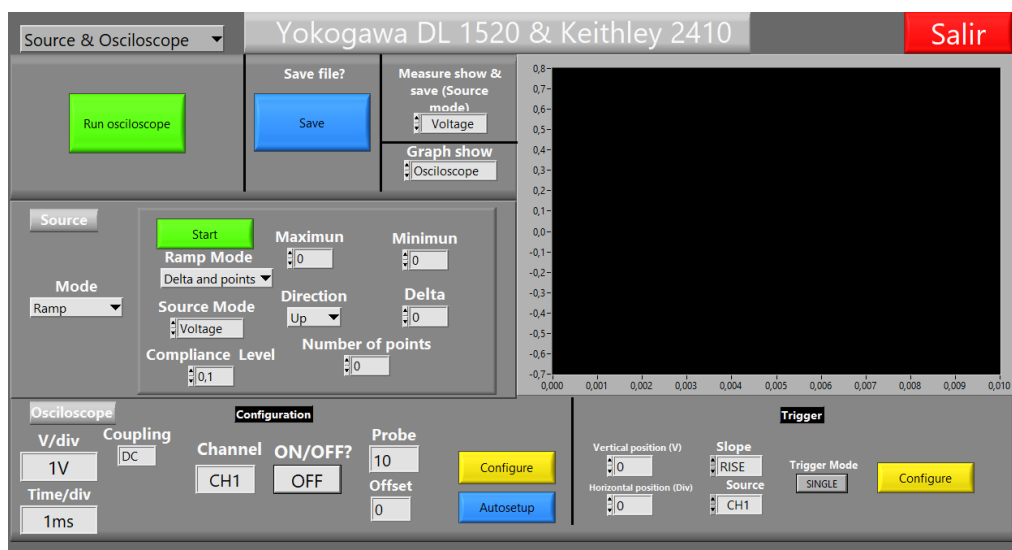


Figura 4.1 Pantalla principal

Este es el aspecto principal que tiene la aplicación al iniciarse. Vamos a diferenciar dos modos de trabajo, los dos instrumentos que tenemos.

Como comentamos en el apartado donde explicamos el osciloscopio que utilizamos, vamos a detallar los diferentes controles que tenemos en la aplicación para este instrumento.

4.1.2. El osciloscopio

Lo primero que vamos a pasar a describir son los controles del osciloscopio, para ello vamos a describir uno a uno cada control y la función que realiza.

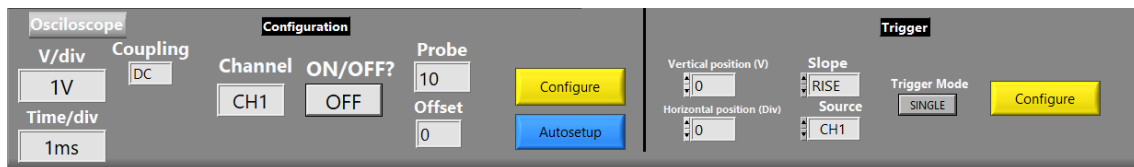


Figura 4.2 Detalle de la parte inferior

Como se ve en la imagen se diferencian dos zonas de configuración, una para los parámetros básicos y otra para el trigger.

4.1.2.1. Configuración de parámetros pantalla.



Figura 4.3 Menú de configuración de osciloscopio

Pasamos a comentar los parámetros que podemos modificar para la visualización de las señales en el osciloscopio. Aquí tenemos la opción de V/div, para cambiar el tamaño del eje y del osciloscopio. Otra opción es el Time/div, para modificar el valor del eje x, pudiendo ser modificado entre 50 segundos a 2 nanosegundos por cada división. Otro parámetro configurable es el “Coupling” que nos permite visualizar que parte de la señal queremos ver, a elegir entre la componente AC, la DC o GND. Adicionalmente se puede elegir si se desean ver los dos canales simultáneamente o por el contrario si se quiere ver cada canal individualmente. Por último se permite elegir también si se quiere una atenuación a la entrada de la señal, esto se hace mediante la opción “Probe” y en último lugar si se quiere un offset para mover toda la señal el valor que se quiera aquí.

Para que todos estos parámetros sirvan para algo y el osciloscopio realice lo deseado, se ha de pulsar el botón “Configure”. Esto lo que provoca es una actualización de la configuración en el osciloscopio que se verá reflejado cuando se encuentre en modo “Start”.

También se puede hacer una autoconfiguración del osciloscopio. Esta opción nunca es recomendable ya que la señal puede cambiar con el tiempo por tanto se perdería información si este modo se activa de forma habitual. Puede ser útil si se ha perdido la

configuración manual o no se encuentra la señal, pero lo recomendable es a partir del “Autosetup”, configurar los parámetros al gusto.

4.1.2.2.El trigger

La segunda zona que se diferencia de la configuración del osciloscopio es la relativa al trigger.

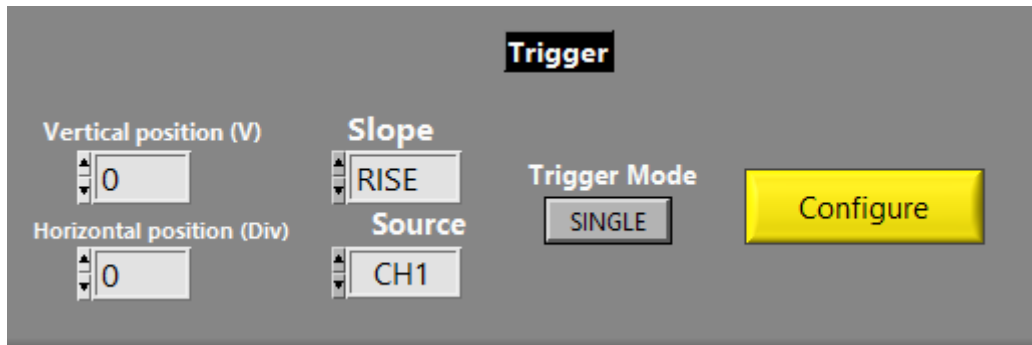


Figura 4.4 Menú de configuración del trigger

Aquí la configuración es simple. Se tiene dos opciones para la configuración vertical y horizontal. La vertical se rige por el valor numérico de la señal a medir, es decir, se tiene en cuenta el valor del voltaje. Por el contrario, la posición horizontal se basa en el valor de las divisiones horizontales del osciloscopio y no en el tiempo que se desee.

Las demás opciones son el tipo de flanco que se desea utilizar, si de subida, bajada o ambas. Así como de donde se quiere que se coja la información para el trigger. Al igual que para la pantalla para actualizar esta información es necesario pulsar el botón configurar para mandar la información al osciloscopio.

4.1.3. La fuente

Ahora pasamos a comentar las opciones que tenemos en la aplicación para la configuración de la fuente. Aquí vamos a hacer una diferenciación, ya que como comentamos cuando describimos el osciloscopio, hay varios modos de generación de señales. Vamos a diferenciar dos modos, rampa y constante.

A la izquierda tenemos un botón de selección, llamado “Menu Ring” en LabView. Este selector es común en ambos modos. La diferencia recae en que, dependiendo del modo seleccionado, cambia la parte derecha.

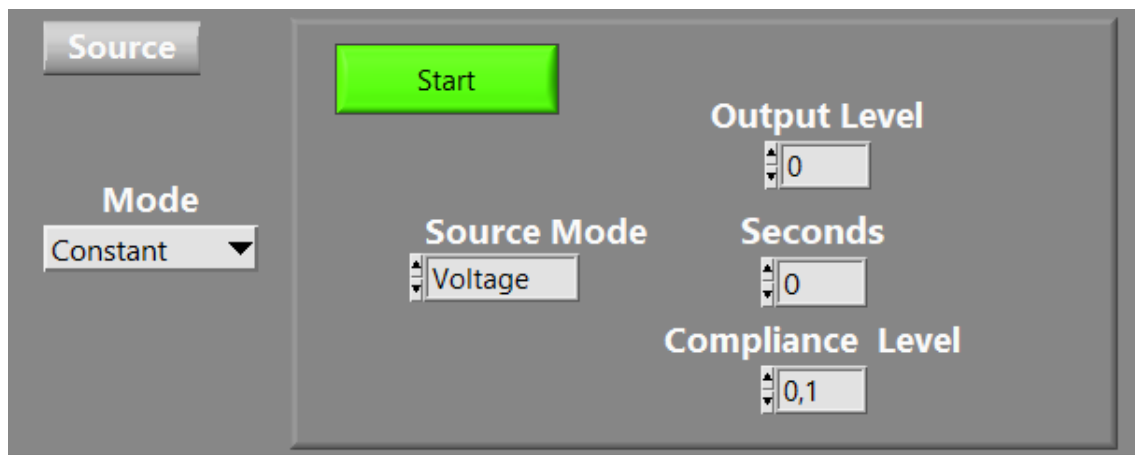
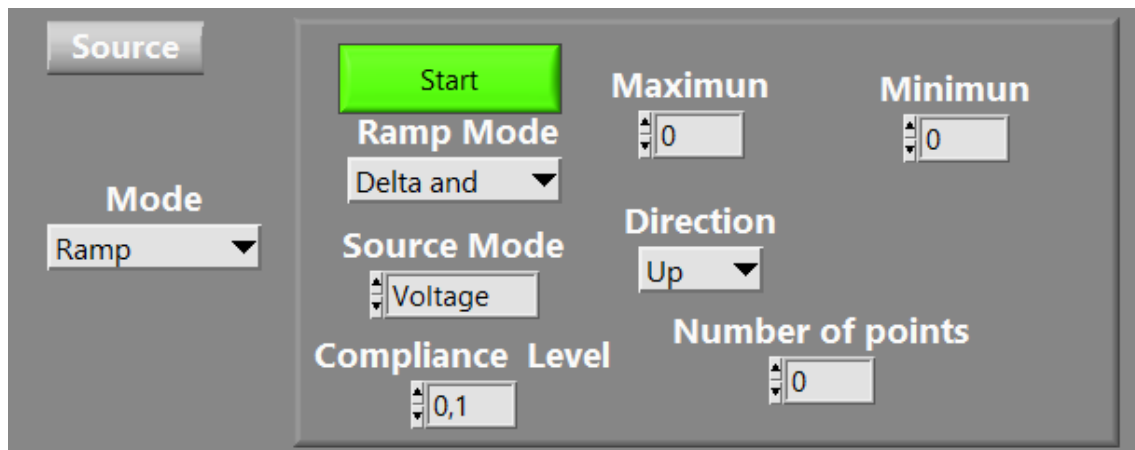


Figura 4.5 Detalle de configuracion de la fuente (Modo rampa, arriba, y modo constante, abajo)

4.1.3.1. Modo rampa

Comenzamos con el modo rampa que hemos programado para poder generar señales lineales ascendentes o descendentes. Al igual que la selección de modo rampa o constante aquí también tenemos dos submodos. El modo “Delta & Points” y el modo “Normal”. Como vemos en la imagen inferior, existe otro “Menu Ring” con esas dos opciones.

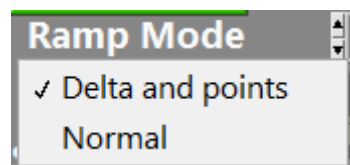


Figura 4.6 Detalle de modos de rampa

Dependiendo de la opción seleccionada, tenemos dos modos de funcionamiento diferentes, pero con muchas similitudes. La gran diferencia recae en el “delta” de tiempo entre cada punto. Ambos modos tienen los mismos parámetros de configuración

exceptuando el parámetro “Delta”. Este valor, disponible solo en su modo, permite el espaciado entre puntos. Por el contrario, en el modo normal, este espaciado es automático, es decir, la fuente genera los puntos a la mayor velocidad disponible.

Ahora bien, los demás parámetros disponibles son los habituales para poder hacer una rampa. Existe el valor máximo y mínimo de la señal, la dirección que sigue la rampa, ascendente o descendente, el número de puntos intermedios de la señal y el valor de compliance, es decir, como ya comentamos en la descripción del equipo, el valor de intensidad máxima de salida.

También es posible elegir si la fuente excita con corriente o con voltaje. Con “SourceMode” se consigue este efecto.

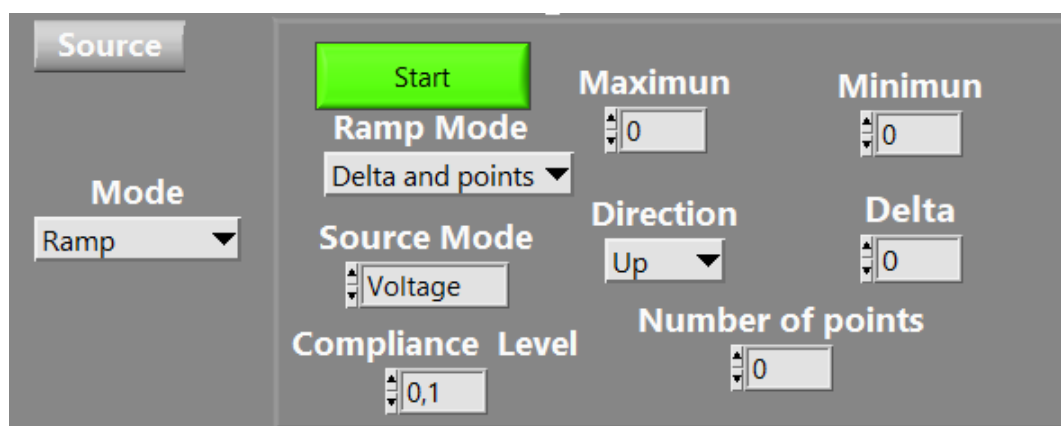


Figura 4.7 Aspecto modo rampa

4.1.3.2. Modo constante

El modo de salida constante es útil cuando queremos excitar una muestra con una tensión continua. Para ello tenemos el siguiente modo.

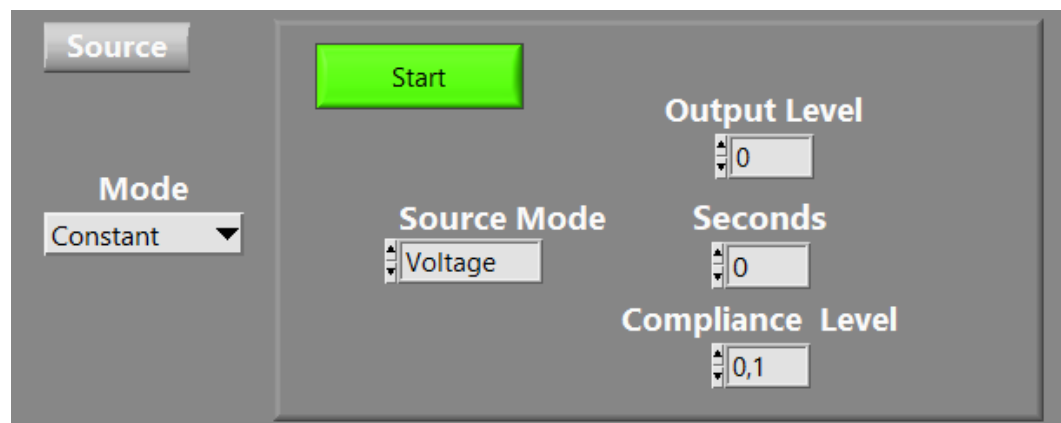


Figura 4.8 Detalle modo constante

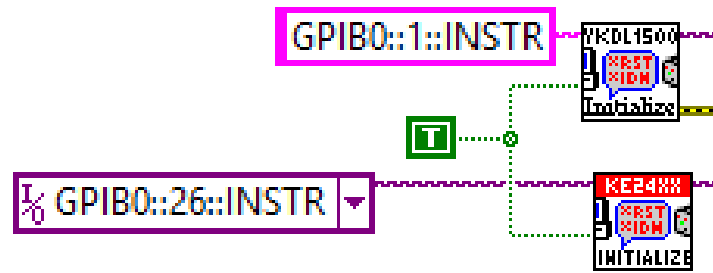


Figura 4.10 Detalle de subVIs de inicialización

Si nos fijamos en detalle en la primera parte, vemos los subVIs de inicialización en primer lugar. Estos VI abren la sesión del osciloscopio y la fuente. La dirección GPIB de ambos es configurable en cada instrumento, pero para este caso hemos dejado el osciloscopio con la dirección 1 y la fuente con la 26, que eran la que tenían por defecto por lo que se decidió no cambiarlo.

Una vez la sesión esta inicializada hacemos una pequeña secuencia de configuración rápida de algunos parámetros.

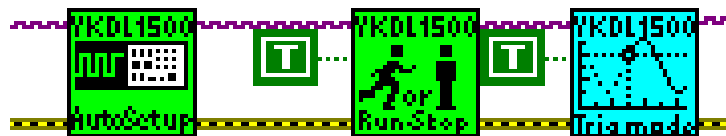


Figura 4.11 Secuencia de inicialización

En primer lugar, se hace un “AutoSetup”, para poder recuperar la visualización de la señal en caso de que estuviera mal configurada y no se viera con claridad en la pantalla. Posteriormente se arranca el osciloscopio, es decir, se pone en modo “Start” para poder visualizar la señal en la pantalla del osciloscopio. Y posteriormente se configura el modo del trigger para la visualización en modo automático y así poder comenzar a ver la señal bien colocada.

Como resumen de esta secuencia, podemos decir que tiene la finalidad de comenzar con la configuración con una señal que poco a poco podemos ir cambiando. Permite una primera aproximación de lo que vamos a hacer.

4.2.2. El desarrollo

Una vez acaba la inicialización, se pasa a los dos bucles que se ejecutan en paralelo y que vamos a diferenciar en el bucle del osciloscopio y en el bucle de la fuente.

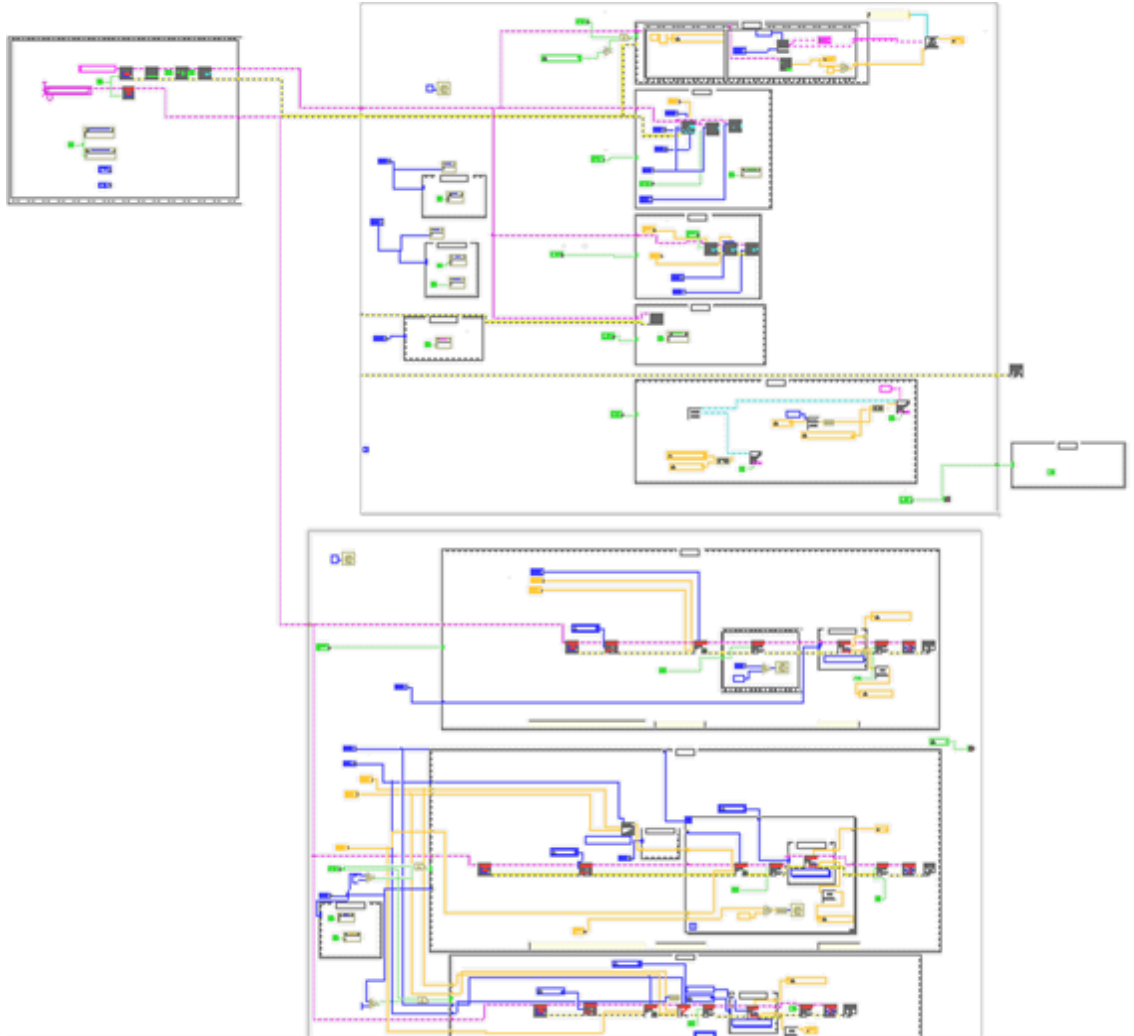


Figura 4.12 Aspecto general del código

La imagen superior nos da una idea de la estructura general del código. LabVIEW no permite hacer zoom a lo largo del código, por lo que la imagen no es la mejor del mundo. Para lograr ver este efecto existe una opción en LabVIEW llamada “Navigation”. Esto abre una ventana que permite ver todo el código en una resolución baja para mediante el ratón situarnos en la parte que deseemos.

Una vez explicado esta limitación que trae el software, en la imagen podemos ver los dos bucles. Son las dos grandes estructuras tras la inicialización y que están conectadas con el hilo morado, color que determina que se trata de una sesión de GPIB abierta.

Si nos fijamos en los dos bucles, que son bucles “While”, diferenciamos el del osciloscopio, en la parte superior, y el de la fuente, en la parte inferior.

4.2.2.1. El osciloscopio

Empezamos describiendo el bucle superior, el relativo al osciloscopio. Vamos a comentar cada botón de acción del modo del osciloscopio.

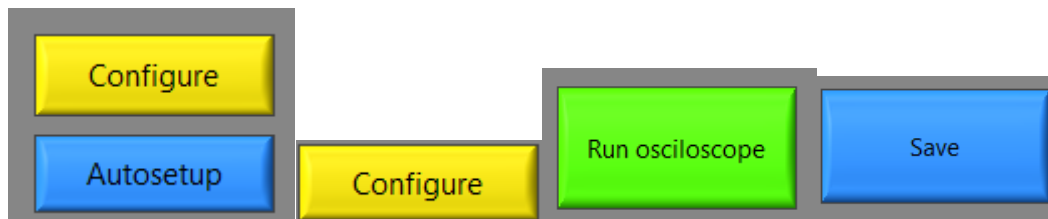


Figura 4.13 Botones de la aplicación

Pero antes de empezar estos botones y sus efectos, vamos a explicar primero las posibilidades que tenemos con el juego de ventanas dependiendo del modo de funcionamiento.

4.2.2.1.1 El control visual

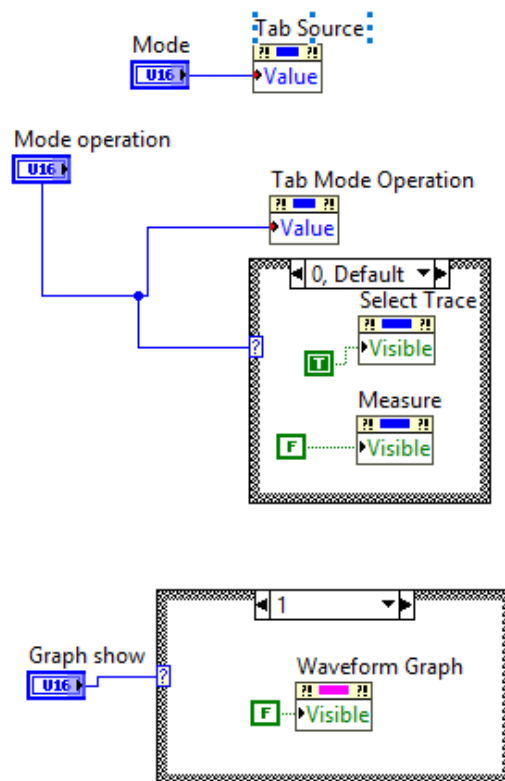


Figura 4.14 Configuración de aspectos de la interfaz

Arriba podemos observar estos dos “Case Structure”. Todos cumplen funciones estéticas de funcionamiento. Estos puntos permiten que, dependiendo de cada opción de funcionamiento seleccionada, se muestren una ventana u otras.

Si vemos el primer control y “Property node”, vemos que dependiendo del selector llamado “Mode”, cambiara la venta de visualización del modo de la fuente, o rampa o constante.

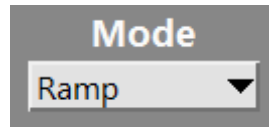


Figura 4.15 "Menu ring" del modo de funcionamiento de la fuente

El selector es un “Menu Ring” con dos opciones, que internamente equivalen a un 1 o un 0. Si el selector está en “Ramp”, el Tab control de los modos de la fuente selecciona el tab correspondiente, o rampa o constante, lo que mostraría las opciones de rampa o constante

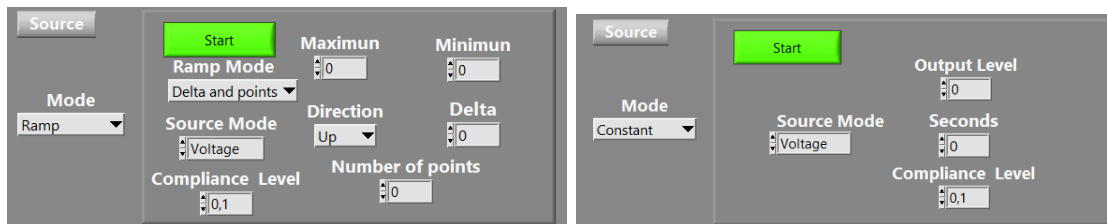


Figura 4.16 Opciones de la fuente

El Segundo control, el llamado “Mode operation”, discrimina los dos modos que tenemos, la utilización de solo el modo osciloscopio o ambas. En caso de que se elija el modo osciloscopio, solo se mostrará el menú del osciloscopio. Por el contrario si selecciona el otro modo se verá las opciones de la imagen anterior.



Figura 4.17 "Menu ring" de modo osciloscopio o fuente con osciloscopio

Al igual que el “Menu Ring” de “Mode”, internamente cada modo equivale a un 0 o un 1. Si se selecciona el modo “Osciloscope” El aspecto será el siguiente.

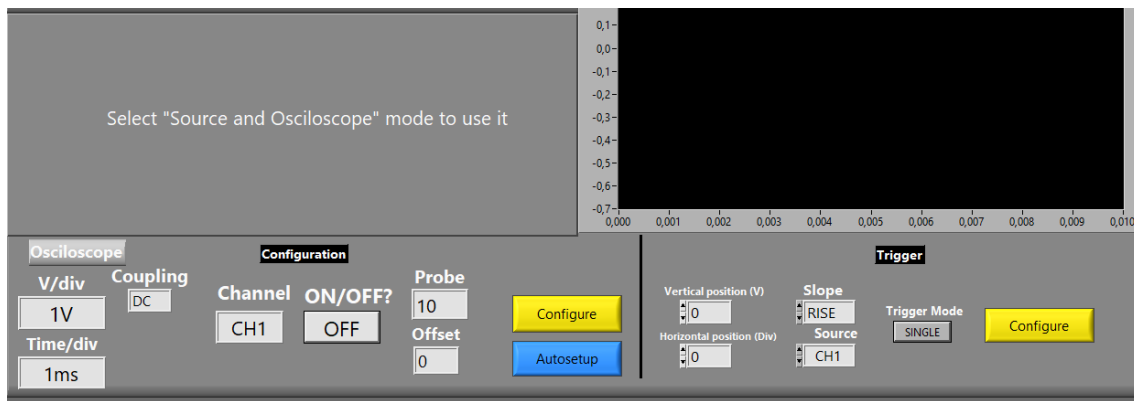


Figura 4.18 Detalle del modo osciloscopio

La pantalla de la fuente desaparece y aparecerá un mensaje advirtiéndole que para poder ver estas opciones se debe cambiar de modo.

Adicionalmente, en la parte superior central aparece otro selector.

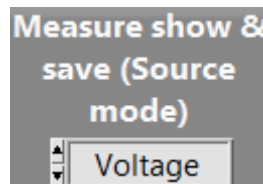


Figura 4.19 "Menu ring" de la gráfica de la fuente

Este control solo aparece en el modo "Source & osciloscopio". Esto permite definir qué datos medidos con la fuente queremos que se guarden. Mas tarde veremos cómo se guardan estos archivos.

4.2.2.1.2 Los botones del osciloscopio

Una vez hemos explicado la interfaz visual veamos como está programado cada uno de los botones de confirmación de opciones de configuración. Comenzamos con los botones de configuración de la visualización de los datos en la pantalla del osciloscopio.

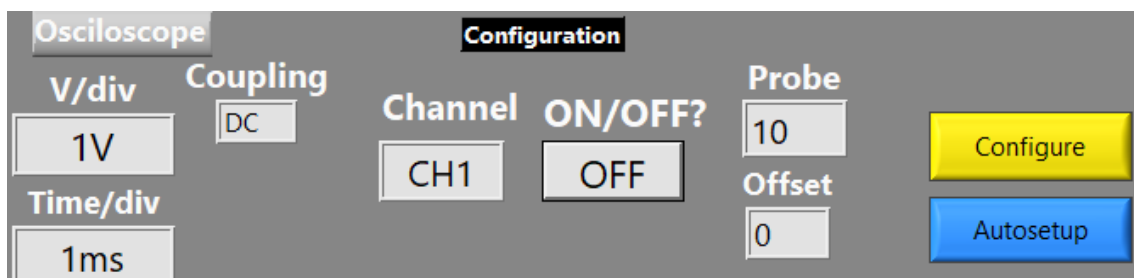


Figura 4.20 Configuración del osciloscopio

Arriba vemos dos controles, “Configure” y “Autosetup”. El botón “Configure” realiza la secuencia siguiente.

- “Configure”

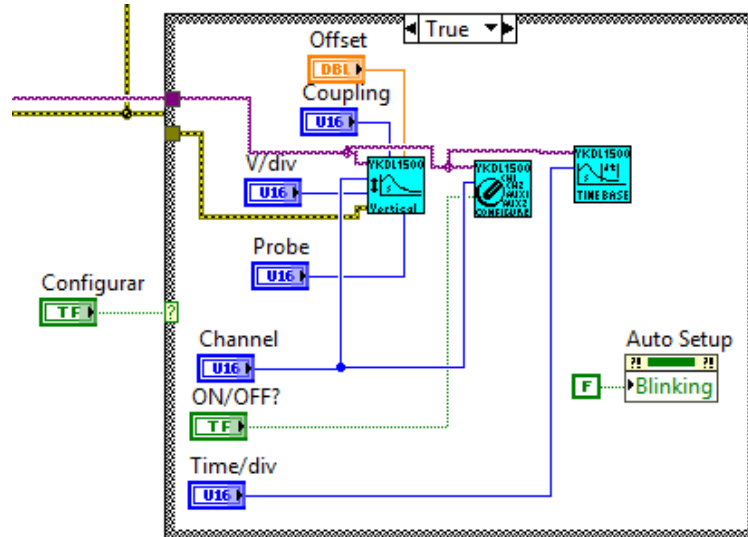


Figura 4.21 Programación del osciloscopio

Aquí vemos el “Case structure” del botón configurar. Existen siete parámetros de configuración. Anteriormente vimos cada control en el panel frontal, pero ahora lo vemos en el diagrama de bloques. Diferenciamos tres subVIs, la configuración vertical de la pantalla, la opción de activar o desactivar cada canal y la configuración vertical.

En primer lugar, al subVi de configuración vertical se introducen los datos del canal a configurar, la atenuación de la señal, los V/div, el offset deseado y el “coupling”, para poder ver la señal en alterna o solo la componente de continua.

En el segundo VI solo se introduce el número de canal y si queremos activarlo o desactivarlo. Y por último en el tercero solo se configura el tiempo por división del osciloscopio.

Como extra vemos un “Property node”. Este “Property node” está referido al botón de “AutoSetup”. Como veremos ahora, el botón de “AutoSetup” parpadea intermitentemente para indicar que se está ejecutando ese modo. Al poner este “Property Node” a falso, al pulsar el botón de configurar manualmente el osciloscopio, desactivamos el indicador intermitente.

- **“AutoSetup”**

Pasamos a comentar el “AutoSetup”. Este control ejecuta automáticamente la configuración automática de la pantalla del osciloscopio.

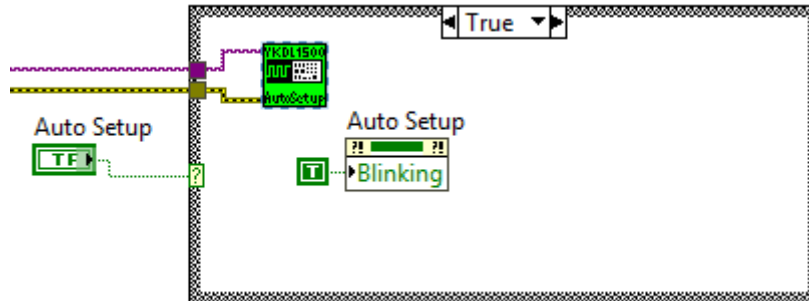


Figura 4.22 Programación "AutoSetup"

La programación aquí es simple, únicamente hay que llamar al subVI para su ejecución. Como ejemplo de la programación SCPI vemos como es el subVI interiormente. Únicamente es una comunicación VISA al que mandamos el comando correspondiente. Mas adelante veremos algún ejemplo más complicado.

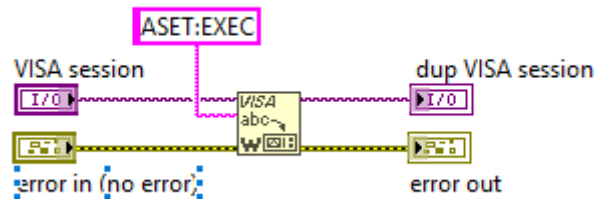


Figura 4.23 Comando SCPI para AutoSetup

- **“Configure trigger”**

El siguiente control que vamos a comentar es la configuración de trigger.

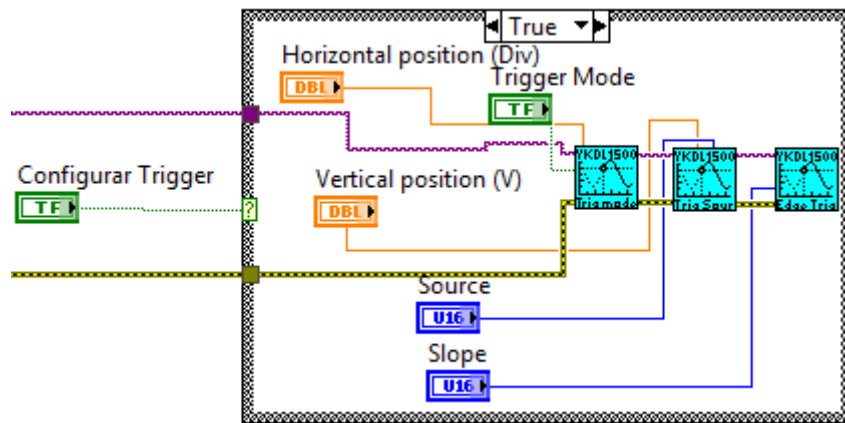


Figura 4.24 Programación del trigger

En este caso tenemos cinco parámetros diferentes y tres subVIs. Aunque los tres subVIs tengan la misma imagen, son totalmente diferentes. El primero tiene por finalidad la selección del modo de disparo, ya sea el modo normal o el modo “single” que solo permite ver una toma de medidas antes de que el osciloscopio se pare. También se introduce el valor de la posición horizontal que se desee, es decir, el offset horizontal que queremos ajustar.

Con este VI podemos entrar en más detalle en su interior.

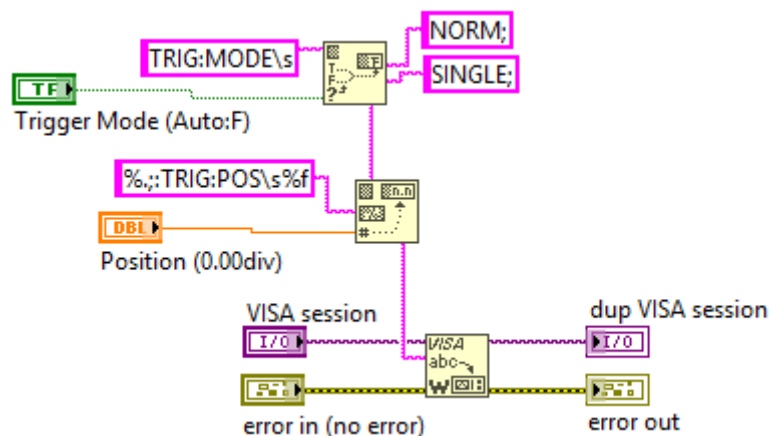


Figura 4.25 Detalle de comando SCPI para configuración del trigger

Vemos como arriba dependiendo del modo seleccionado, se crea un string con el modo normal o el single. Posteriormente se concatena este string al formado con la posición horizontal que se desea y posteriormente, ya creado el comando SCPI se manda mediante la sesión VISA abierta.

Por otra parte, el segundo VI sirve para decidir que canal es el que va a actuar como disparo y que offset vertical deseamos. Y por último tenemos el modo de detección del trigger, si con flancos de subida, bajada o ambos.

- **“Run oscilloscope”**

Ahora pasamos a un comando algo más complicado. Para el funcionamiento correcto de la toma de datos y de la visualización de la gráfica en la aplicación es necesario varios pasos. Vamos a ir explicándolo poco a poco.

Para poder visualizar la señal en la aplicación es necesario que el osciloscopio este parado. Para conseguir este efecto debe primero configurarse el trigger en modo “single”, como se ha explicado anteriormente. Una vez se ha configurado el trigger en este modo y vemos que la pantalla en el osciloscopio está parada, al pulsar en “Run oscilloscope”, la misma señal se verá reflejada en la aplicación

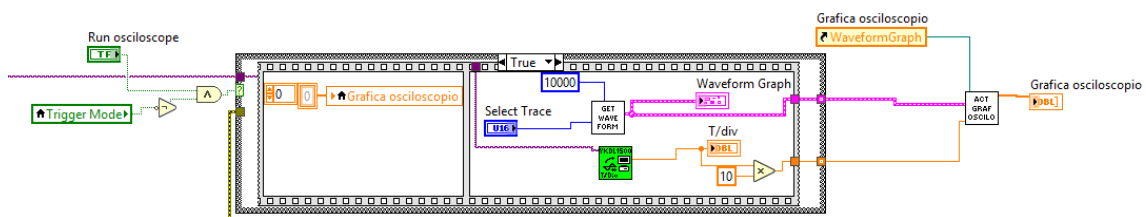


Figura 4.26 Programación de "Run Oscilloscope"

Ya una vez se ha hecho esa secuencia pasamos a describir que ocurre cuando pulsamos en botón de “Run oscilloscope” y el modo de trigger es “single”. Dentro del “Case structure” se coloca una “Flat sequence”. Primero tenemos que reiniciar el valor de la gráfica para borrar los datos anteriores. Después lo que hacemos es llamar a un subVI (en blanco y negro en la imagen) con el que vamos a devolver los datos de la señal obtenidos con el osciloscopio.

A este subVI lo que hacemos es pasarle dos parámetros, el canal del cual queremos ver la señal y una constante de valor “10000”. Este valor es el número de puntos que vamos a obtener; con este valor tenemos una resolución suficiente.

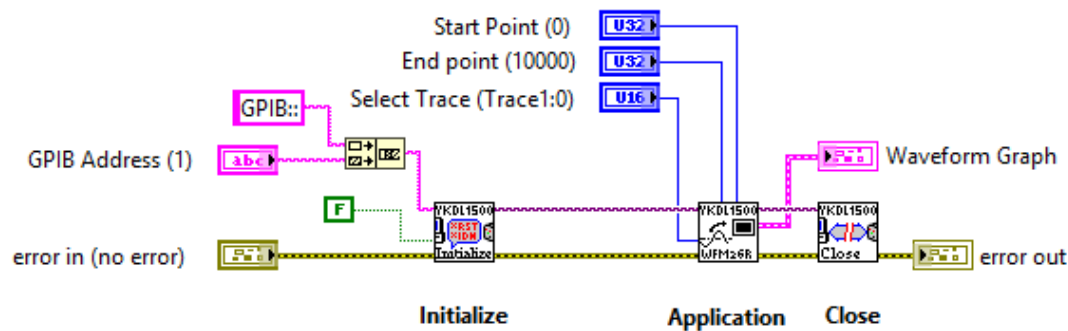


Figura 4.27 Programación de la toma de datos del osciloscopio

En la imagen superior vemos que contiene internamente el subVI. Lo primero es inicializar la sesión, como ya comentamos anteriormente con la dirección GPIB número 1. Después llamamos a otro subVI al que pasándole los parámetros del número de puntos y el canal, nos devuelve la gráfica para su tratamiento posterior.

Volviendo a la secuencia principal, una vez tenemos la gráfica para poder mostrarla (en color rosa), se guarda en un indicador y se muestra en pantalla. Si nos fijamos hay otro subVI en color verde. Este únicamente lo que hace es obtener el valor del tiempo por división configurado en el osciloscopio.

Al acabar la secuencia lo que se hace es llamar a otro subVI para poder guardar los datos en un fichero externo. Este VI lo que tiene que hacer es transformar la gráfica a un array de enteros para poder tratar los datos.

A este subVI se le manda la gráfica y el valor del tiempo por división obtenido anteriormente. Lo que nos devuelve es el valor de la gráfica en enteros en un array.

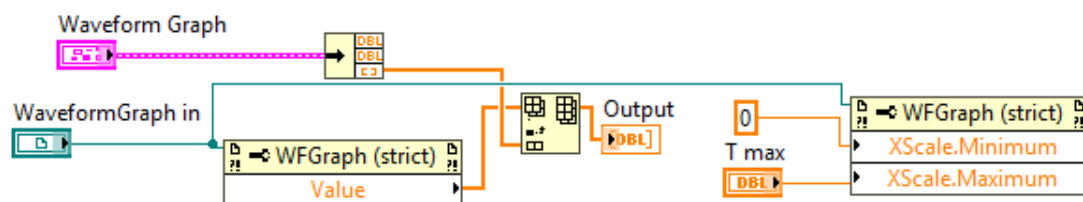


Figura 4.28 SubVI de transformación de la gráfica

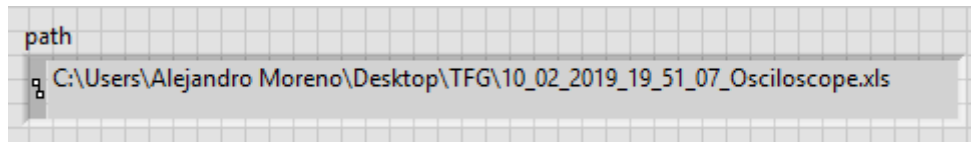


Figura 4.31 Ejemplo de path creado

Se configura día, mes, año, hora, minuto y segundo junto con la extensión .xls para visualizarlo en Excel. Se crean dos directorios, uno para guardar los datos del osciloscopio y otro para la fuente.

Una vez tenemos creados los directorios del salvado de datos lo que hacemos es pasarlo al subVI proporcionado por LabVIEW para guardar los datos. Para el guardado de datos del osciloscopio es necesario obtener los datos de tiempo de cada dato guardado. Para ello tenemos un subVI que creamos al que le pasamos el dato del tiempo por división obtenido del osciloscopio y el número de puntos.

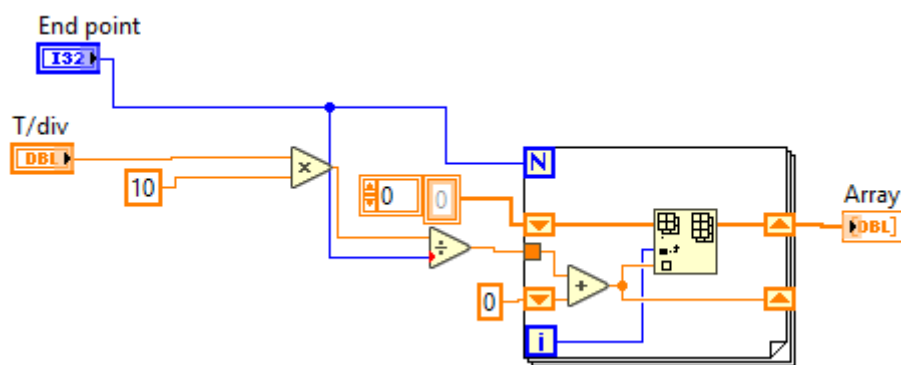


Figura 4.32 Creación array de datos para guardar

Lo que se hace es multiplicar por diez el tiempo por división, ya que el osciloscopio tiene diez divisiones en la pantalla, así tenemos el tiempo total, que dividido por el numero de puntos, obtenemos el valor de tiempo de cada punto.

Una vez se tiene el tiempo creamos un array de dos dimensiones con los datos y el tiempo. Que pasamos al VI de LabVIEW, llamado “Write Delimited Spreadsheet”.

4.2.2.2. La fuente

Ya cerrado el capítulo del osciloscopio, pasemos a describir la parte relacionada con la fuente. En esta parte del código se va a estructurar el código de forma secuencial. Los pasos que se siguen para la realización de las rampas o la constante siempre siguen una estructura marcada con pasos.

4.2.2.2.1 Los botones de la fuente

En este punto vamos a comentar los dos botones de “Start” que se sitúan dentro del “Tab control” de la fuente.

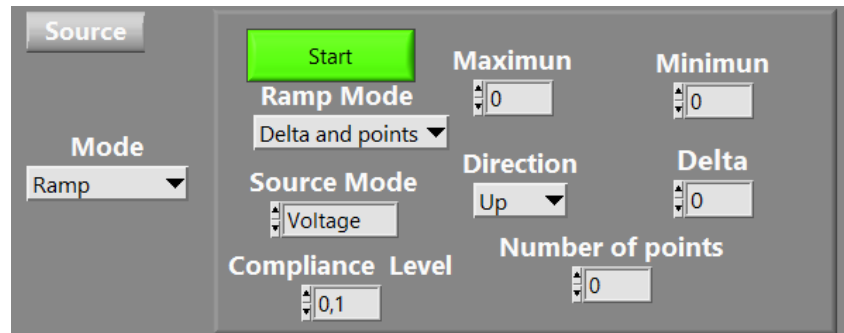


Figura 4.33 Detalle modo rampa de la fuente

Como ya hemos comentado anteriormente, al seleccionar el modo de funcionamiento veremos dos pantallas, correspondientes a la rampa y a la constante.

- **Constante**

Este modo como su nombre indica y a hemos comentado antes proporciona una salida constante durante un tiempo determinado.

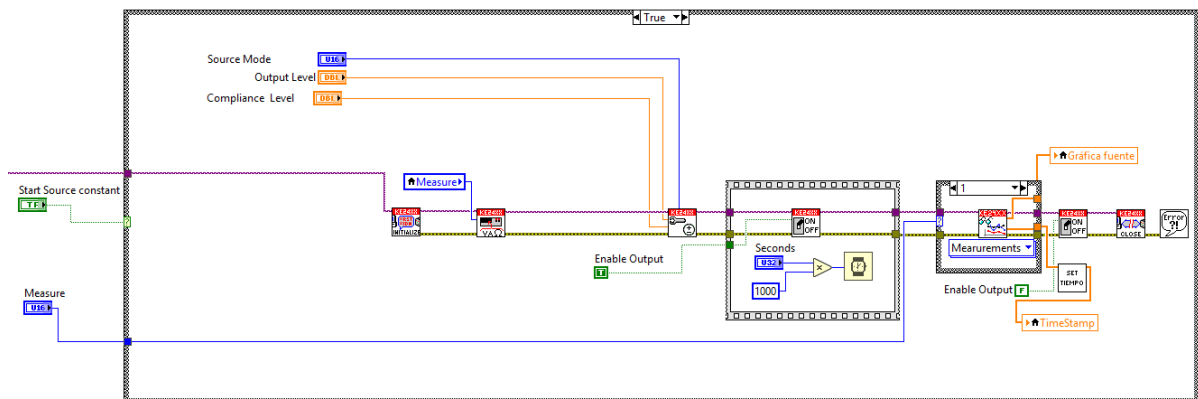


Figura 4.34 Programación de constante

Arriba vemos la secuencia que tiene esta parte del código. Vamos yendo por orden de subVIs. El primero que tenemos es el de inicialización. Aquí lo único que se hace es comprobar que el dispositivo está conectado, que se encuentra libre para poder ser usado y que están todos los parámetros listos para comenzar.

El siguiente subVI es una ayuda visual que tendremos a través de la pantalla de la fuente. En el caso de la imagen aparece un “Measure”. Esto quiere decir que en la pantalla se podrá observar el valor de la medida que esté tomando nuestro dispositivo. Son configurables varios parámetros para poder verse a través de la pantalla.

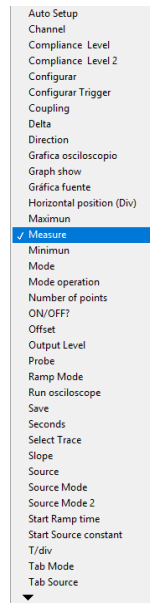


Figura 4.35 Ejemplo de configuraciones de pantalla posibles

El siguiente subVI es el encargado de configurar la constante queremos. A este subVI le pasamos el modo de funcionamiento que queremos, o tensión o corriente, el valor de dicho parámetro y límite de corriente que tendremos a la salida.

Posteriormente tenemos una “Stacked sequence” que está temporizada por el valor de tiempo que queremos habilitar la salida. Este valor se introduce en el panel frontal como segundos, pero el temporizador que tenemos en el interior lo lee en milisegundos, por lo que es necesario multiplicarlo por mil.

El subVI que se encuentra en el interior es el encargado de activar o desactivar la salida. Sabremos si está activado si cuando llegue a este paso, la fuente emite un pitido y se enciende una luz azul en el frontal.



Figura 4.36 Fuente en funcionamiento

Esa es la señal de que está en funcionamiento el dispositivo.

- **Rampa**

Aquí de nuevo vamos a diferenciar dos aplicaciones de este modo diferentes.

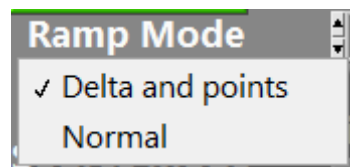


Figura 4.37 "Menu ring" del modo rampa

- Delta and points

De nuevo provocamos que, cambiando entre estos dos modos, visualmente dejemos de ver un parámetro, el tiempo delta entre puntos.

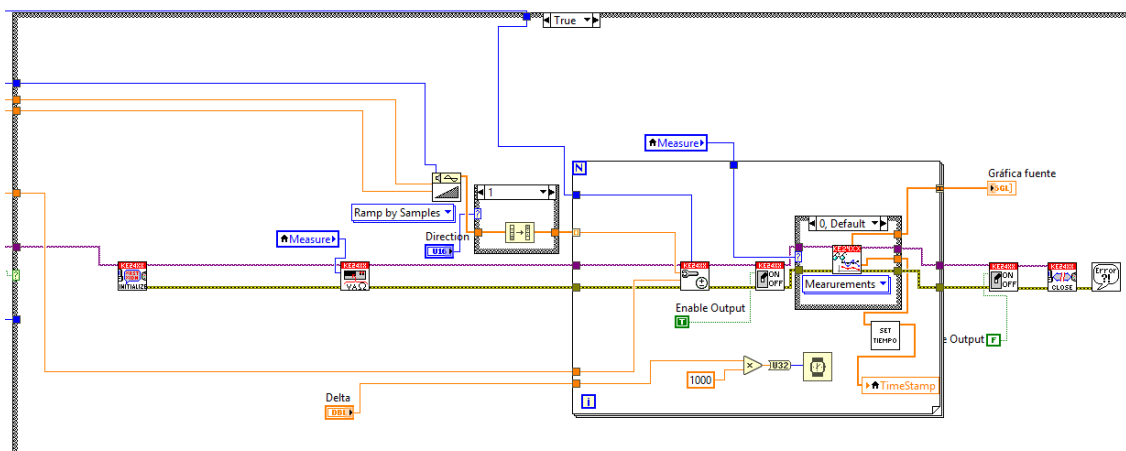


Figura 4.38 Programación de modo rampa

Arriba vemos la secuencia de pasos para el modo de “Delta & points”. Este modo ya los hemos explicado anteriormente, pero ahora vamos al interior. Lo primero que se hace, al igual que en el modo constante, es la inicialización de la fuente y de nuevo configurar para que en la pantalla se muestre el valor de la medida.

En la parte central se observa un subVI que proporciona National Instruments. Este subVI es el encargado de crear un array de puntos con varis parámetros de entrada. De arriba abajo, primero en azul, el número de puntos requeridos, el valor final y el valor inicial. También hay que configurar este subVI como “Rampa por muestras”. Posteriormente encontramos un “Case structure” que tiene dos opciones a elegir, o rampa de bajada o de subida. Si se elige rampa descendente, se invierte el array para lograr este efecto.

Después se observa un bucle “For”. Este bucle se realiza el mismo número de veces que puntos hemos introducido. Para esto, el bucle detecta con el terminal del array el número de elementos que contiene, lo que provoca que el bucle se realice este número de veces cogiendo en cada iteración un valor del array.

Una vez sabemos esto, pasemos a describir el bucle. Al empezar se configura la salida con el primer subVI. Se introduce el tipo de salida, corriente o voltaje, el valor de “Compliance” y se desindexa el array creado con los puntos. Posteriormente se activa la salida, y se toma la medida. Esta medida será mostrada en las unidades que se seleccione, tensión, corriente o resistencia.

Este bucle tiene un temporizador, que es el parámetro delta seleccionado. Cada delta de tiempo configurado, la fuente proporcionará el valor durante ese tiempo el valor del array correspondiente a esa iteración. Recordemos que todos los subVIs y elementos de LabVIEW, salvo que se indique lo contrario, están preparados para aceptar milisegundos como unidad de tiempo.

Se observa un pequeño subVI en color blanco llamado “Set Tiempo”.

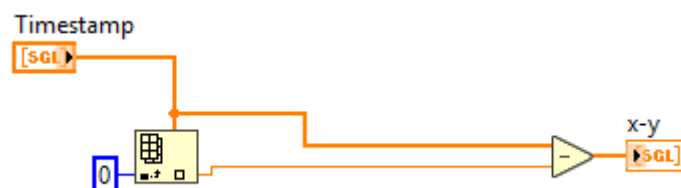


Figura 4.39 Programación rectificación de puntos

Este subVI está preparado para poder obtener una escala de tiempo fiable. Lo que se hace es coger el valor del tiempo que devuelve el subVI de medida, formado por un array, y se le resta el valor de la primera medida. Con esto lo que hacemos es conseguir que la primera medida de tiempo sea el instante cero.

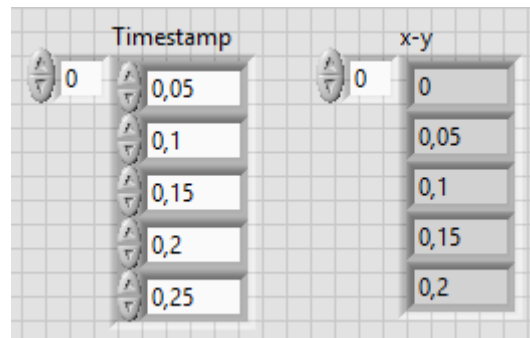


Figura 4.40 Rectificación de los puntos

Ya una vez se hace este bucle “For”, se forma un array con los valores medidos en cada iteración del bucle, lo que hace que esto, junto con el valor del tiempo, se forme un array de dos dimensiones con cada valor medido en su tiempo correspondiente.

Para finalizar el modo rampa “Delta & points” se desconecta la salida de la fuente y se cierra la sesión.

- **Normal**

En este otro modo vamos a hacer solo un pequeño cambio respecto al anterior. Aquí el delta se va a eliminar y el tiempo entre punto y punto va a ser el máximo al que puede llegar la fuente.

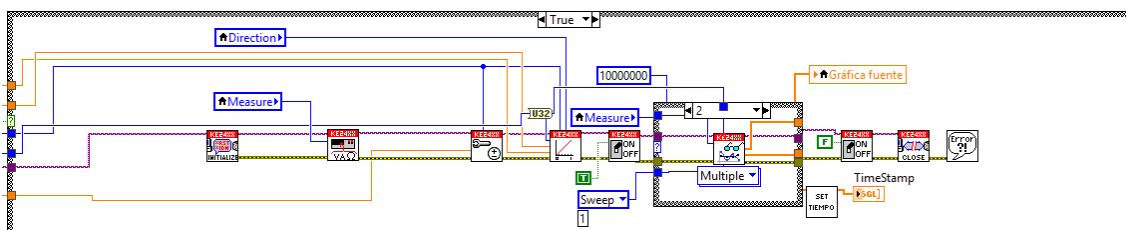


Figura 4.41 Programación modo normal

De nuevo vemos en este modo una secuencia de uso. Si comparamos este caso con el anterior se aprecia solo una diferencia. Los tres primeros subVIs son idénticos y hace la misma función. El siguiente es la diferencia; este subVI es el encargado de programar la rampa que se va a ejecutar. Este subVI es automático. La diferencia con el modo anterior era que nosotros teníamos que programar el tiempo entre puntos, aquí es automático.

Posteriormente se activa la salida de la fuente y se ejecuta el comando de la rampa. Como hemos dicho anteriormente, esta rampa se ejecuta a la mayor velocidad posible de la fuente. Una vez se acaba la rampa, se retoman los datos medidos y hacemos el mismo tratamiento que antes. Por último, se apaga la salida y se cierra la sesión.

4.3. Ejemplo de uso

Ya conocemos todo lo necesario para poder ver un ejemplo de uso de la aplicación.

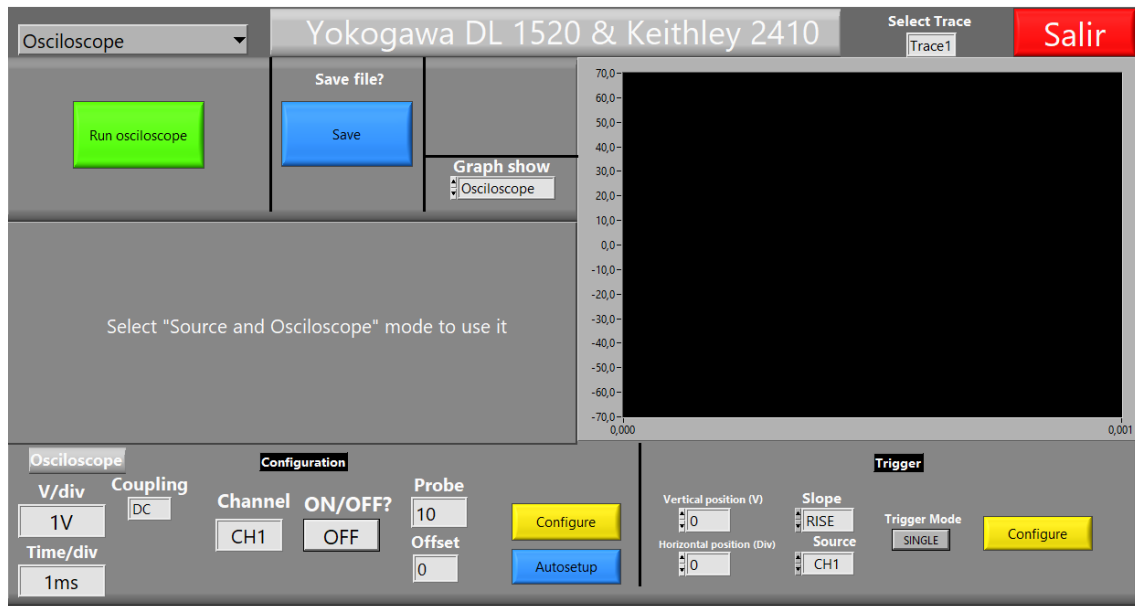


Figura 4.42 Estado inicial de la aplicación

Sabemos cómo luce la interfaz de la aplicación y como se estructura interiormente. Ahora veamos un ejemplo de utilización.

Ya hemos comentado que nada más arrancar la aplicación se hace un auto setup. Visualmente en la aplicación no se observa nada. Hecho el “Autosetup” para poder visualizar lo mismo que se observa en el osciloscopio es necesario parar la señal. Para ello se presiona el botón configure del trigger en modo single. El osciloscopio tomará una medida de la señal que tenga conectada y parará. Posteriormente se pulsa el botón “Run oscilloscope” y anteriormente tenemos que seleccionar el canal que queremos ver con “Select trace”.

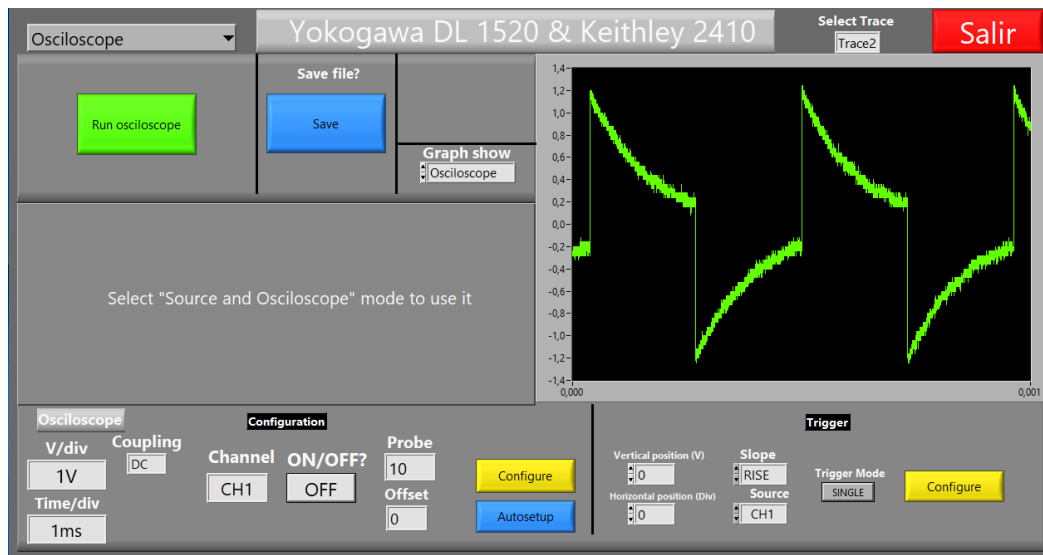


Figura 4.43 Señal mostrada en la aplicación

Arriba vemos un ejemplo del resultado final con un circuito diferenciador. Estimulamos con una señal cuadrada a un condensador y observamos la salida.

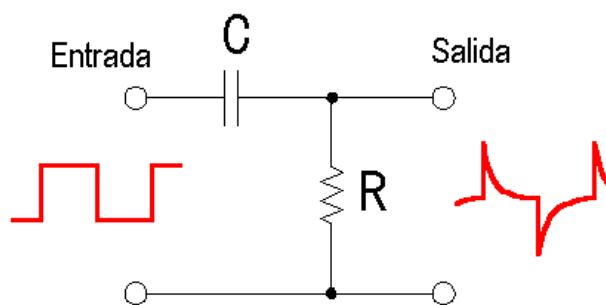


Figura 4.44 Circuito diferenciador [3]

Este comportamiento es el relacionado al osciloscopio. Para la fuente el manejo es más simple.

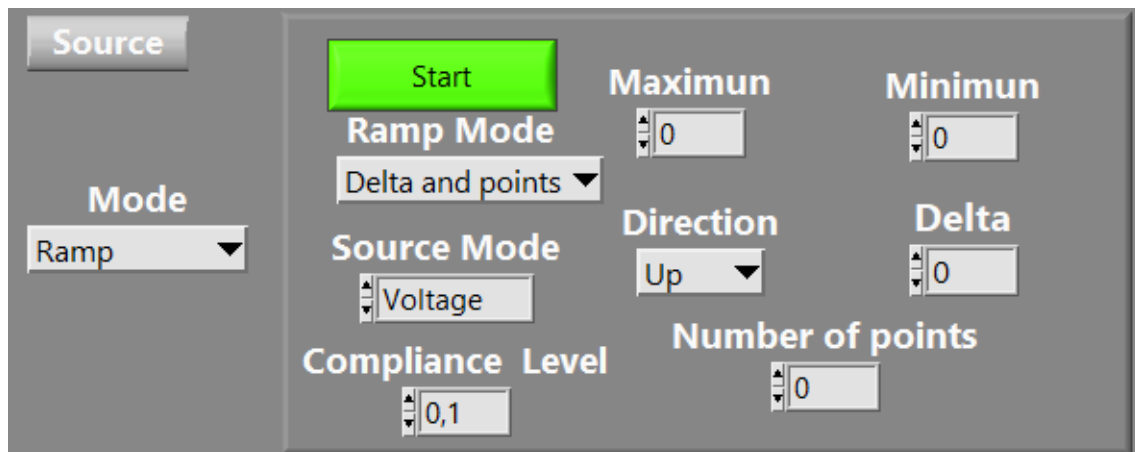


Figura 4.45 Parámetros de modo rampa

Solo es necesario configurar todo como se desea y pulsar “Start”. Para poder ver los datos medido debemos seleccionar mostrar la gráfica de la fuente y el parámetro a medir.

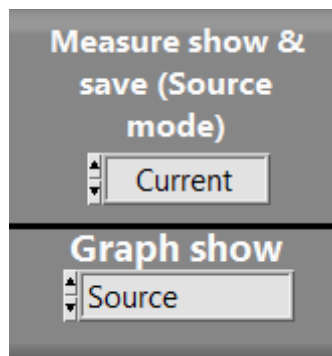


Figura 4.46 Detalle de selección de modos de la grafica

Arriba observamos por ejemplo ver la medida de intensidad de la fuente.

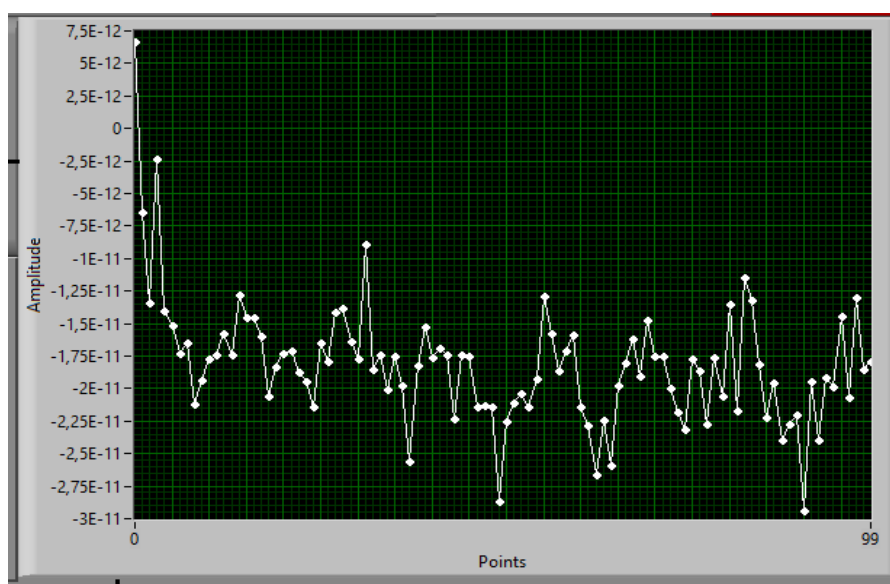


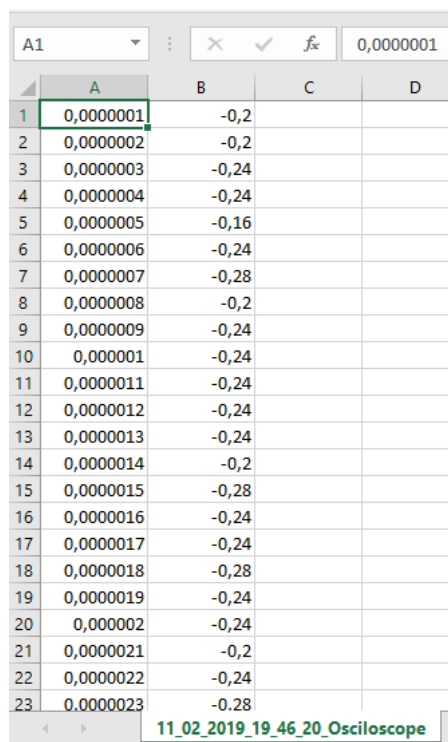
Figura 4.47 Ejemplo de medidas

La gráfica presenta el aspecto anterior.

4.4. Los datos

Un aspecto importante del proyecto es poder guardar los datos de las gráficas en algún fichero externo para su tratamiento posterior. Para este proyecto hemos decidido exportarlo en un fichero Excel que guarda los datos que necesitamos.

Estos datos se guardan en sendos ficheros .xls que se ven así.



	A	B	C	D
1	0,0000001	-0,2		
2	0,0000002	-0,2		
3	0,0000003	-0,24		
4	0,0000004	-0,24		
5	0,0000005	-0,16		
6	0,0000006	-0,24		
7	0,0000007	-0,28		
8	0,0000008	-0,2		
9	0,0000009	-0,24		
10	0,000001	-0,24		
11	0,0000011	-0,24		
12	0,0000012	-0,24		
13	0,0000013	-0,24		
14	0,0000014	-0,2		
15	0,0000015	-0,28		
16	0,0000016	-0,24		
17	0,0000017	-0,24		
18	0,0000018	-0,28		
19	0,0000019	-0,24		
20	0,000002	-0,24		
21	0,0000021	-0,2		
22	0,0000022	-0,24		
23	0,0000023	-0,28		

Figura 4.48 Datos tomados mostrados en una hoja de Excel

La primera columna es el tiempo en segundos y la segunda el valor medido.

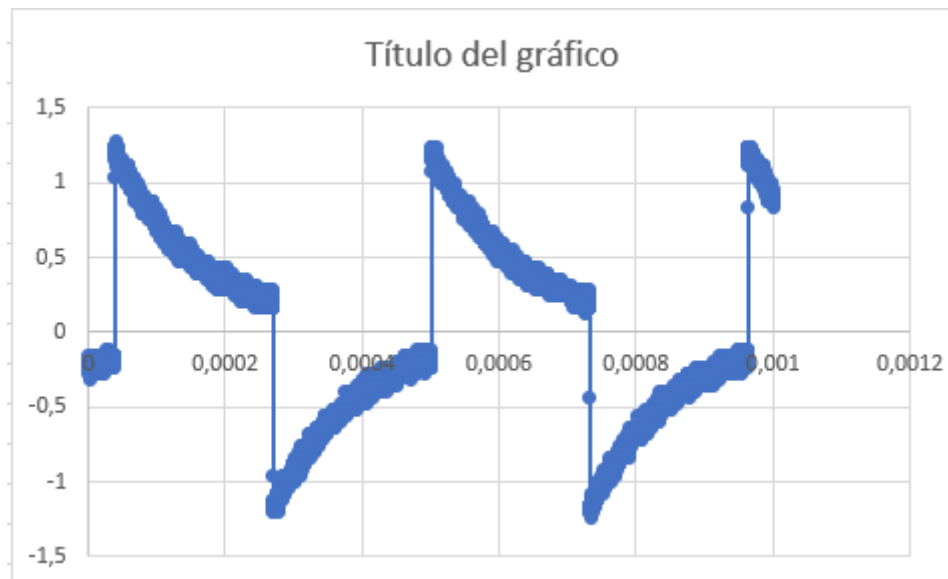


Figura 4.49 Gráfica creada con Excel

Si insertamos una gráfica de Excel observamos lo mismo que en la aplicación. Para el caso de los datos de la fuente la mecánica es la misma, se guardan tantos datos como puntos se configuran.

4.5. El ejecutable y el instalador

Para finalizar el proyecto, el último paso es el crear el ejecutable y el instalador de la aplicación. Esto es necesario hacerlo para el caso que queramos usar nuestra aplicación en otro dispositivo que no tenga instalador LabVIEW.

Antes de nada, es necesario tener todos nuestros subVIs así como el VI principal dentro de un proyecto de LabVIEW.

Nuestro proyecto tiene el siguiente aspecto.

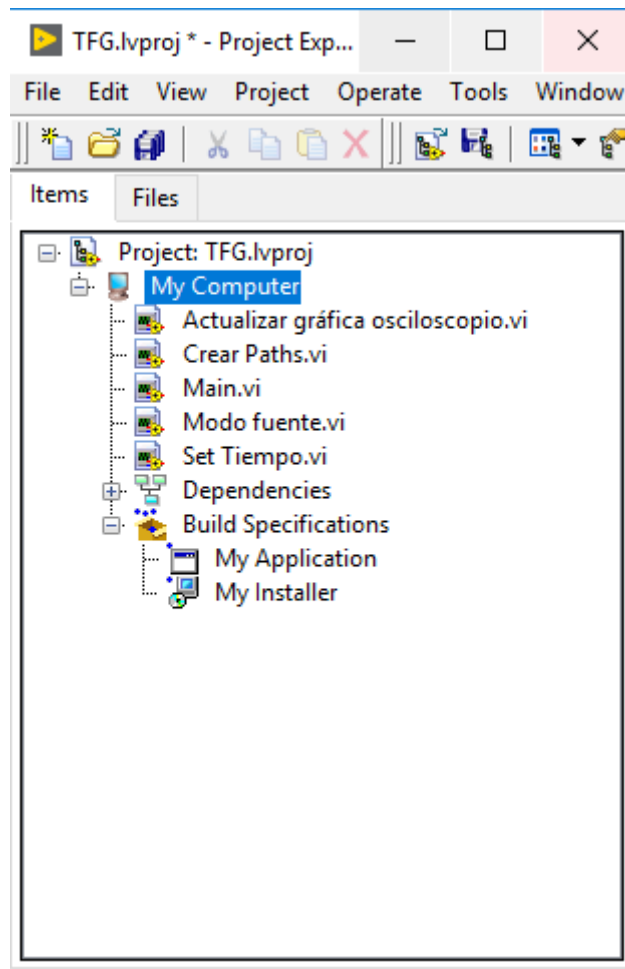


Figura 4.50 Proyecto de la aplicación

Se aprecian los diferentes subVIs del programa así como dos elementos en la parte inferior, “My application” y “My Installer”. Estos dos elementos son lo interesantes en este punto.

Para la creación del ejecutable, “My Application”, es necesario seguir los pasos descritos cuando explicamos como crear un ejecutable. Para ello es necesario explicar cual es el VI principal, al que nosotros hemos llamado “Main”. Este VI se´ra el que se ejecute al iniciar el ejecutable.

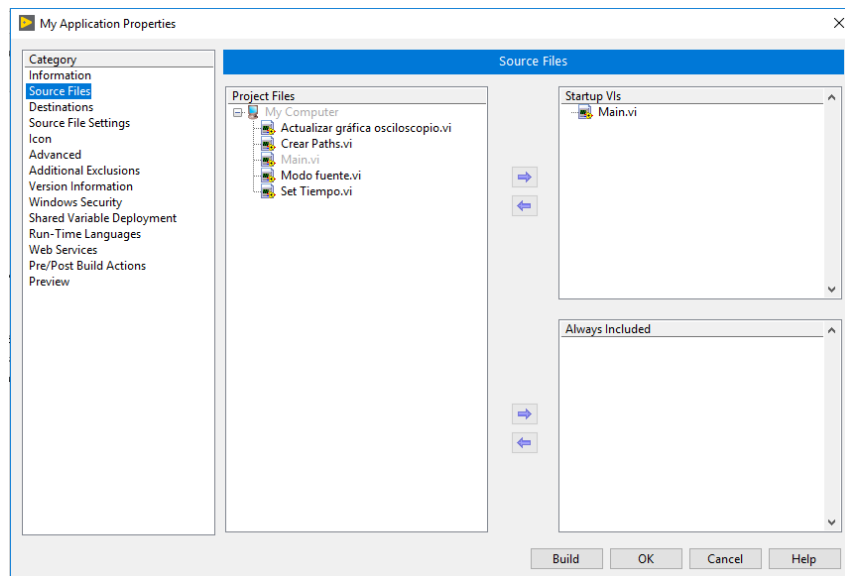


Figura 4.51 Selección de VI principal

Una vez se ha elegido el VI se añaden otros parámetros como el nombre del icono, el símbolo del icono que se verá en Windows o el directorio donde queremos guardarlo.

Para el instalador la mecánica es parecida. Se deciden directorios, nombres e incluso si queremos mostrar mensajes durante el proceso de instalación para ayudar al usuario. Pero lo más importante es seleccionar que ejecutable queremos lanzar con este instalador.

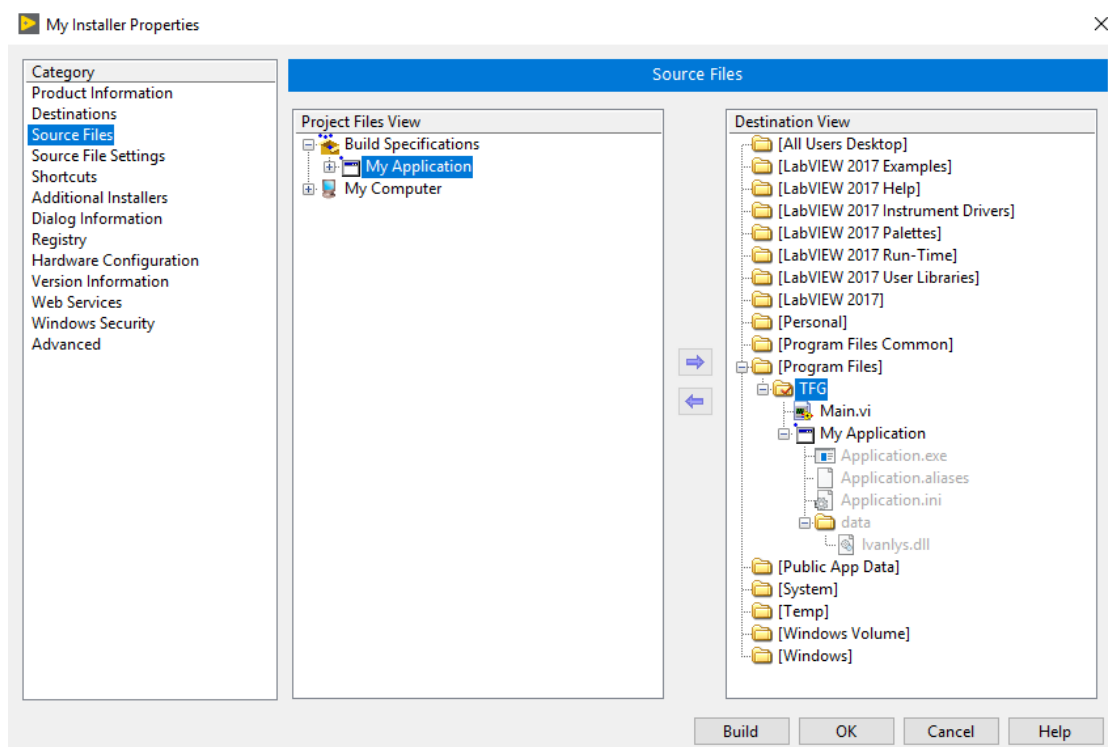


Figura 4.52 Selección de aplicación a lanzar

Es necesario haber creado primero el ejecutable para después, en la ventana mostrada en la imagen superior, seleccionar cual se quiere instalar.

Hay que comentar que es lo que instalamos para que la aplicación pueda ser utilizada en otro ordenador. El responsable de esto es una herramienta de National Instruments llamado NI LabVIEW RunTime. Está herramienta es la que permite ejecutar cualquier ejecutable en cualquier ordenador por lo tanto es que siempre tenemos que seleccionar para guardar en el instalador y poder portarla a cualquier lado.

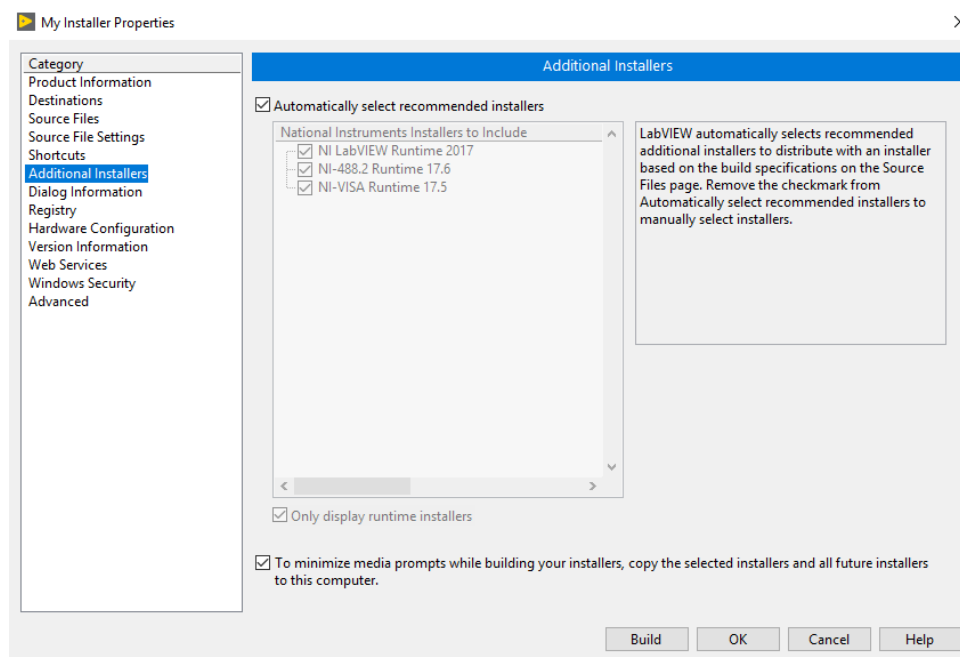


Figura 4.53 Pantalla de selección de programas a instalar

La opción de instalar este complemento es automática, por lo que no es necesario activarla de forma manual. Para este proyecto también se ha tenido que instalar dos complementos más. Uno es el NI-488.2 RunTime y el otro el NI-VISA RunTime. El primero es el encargado de instalar los drivers necesarios en el nuevo dispositivo para poder comunicarnos con GPIB a través del adaptador USB. El otro son los drivers para poder usar todos los subVIs y comandos SCPI de LabVIEW.

Ambos son drivers que han tenido que ser descargados e instalados al comienzo del proyecto. Son drivers de carácter gratuito proporcionados por National Instruments.

5. Conclusiones y futuras mejoras

Durante el desarrollo del proyecto son varias las dificultades y contratiempos que se han ido produciendo para la correcta ejecución del problema. Cabe destacar el primero y más importante.

En un principio el proyecto se iba a desarrollar con una fuente de alimentación diferente. Iba a ser utilizada la Sorensen DSC 150-7. Esta fuente fue descartada para su uso a la hora de empezar a trabajar con ella. En esta fuente era necesaria una configuración hardware externa para poder ser utilizada por GPIB. Esta configuración resultó muy complicada y nunca se llegó a poder comunicar por GPIB. Ante esta problemática y la imposibilidad de, tras intentarlo todo, poder ser utilizada se decidió cambiar de fuente.

Se decidió cambiar a la Keithley 2410 ya que es una fuente que además da más posibilidades de uso. La fuente Sorensen solo iba a permitir una serie de funciones básicas a diferencia de la Keithley que tiene infinidad más de opciones como la de poder tomar medidas.

El proyecto ha tenido más dificultades, esta vez en la parte de programación de la aplicación. Con el osciloscopio el principal obstáculo ha sido el tratamiento de los datos leídos. El objetivo principal era poder visualizar la señal de la pantalla del osciloscopio en la pantalla del ordenador, pero en tiempo real. Se pretendía que se pudiera visualizar cuando el osciloscopio estuviera corriendo, pero esto ha sido complicado de implementar.

También hay que destacar el grabado de datos que no es de lo más eficiente. Los datos no son tratados y solo son mostrados de una forma poco entendible.

Como futuras mejoras se podrían implementar varias adiciones para hacer de esta aplicación una interfaz total de control de los instrumentos. Los controles y opciones menos comunes no han sido utilizados ya que al ser menos frecuentes, se ha priorizado la utilidad y el uso intuitivo antes que la configuración total de todas las opciones.

Se podría haber incluido más opciones relativas al osciloscopio, como limpiar gráficas, configurar medidas o configurar filtros. En cuanto a la fuente igual, hay determinadas opciones que no se han incluido para no hacer el programa muy complejo. También se ha decidido esto ya que la interfaz de la pantalla es limitada y no hay espacio para situar todas las opciones de una forma cómoda de ser utilizada.

Como conclusión del proyecto podríamos comentar que los objetivos planteados al principio de este han sido desarrollados en gran parte. Se ha conseguido una interfaz amigable para el usuario que podrá manejar con mayor facilidad los instrumentos sin necesidad de estar tocando uno a uno.

Se ha conseguido que la programación haya sido posible gracias a los drivers y subVIs proporcionados por los fabricantes, aunque estos subVIs por si solos no tienen ninguna utilidad, hay que usarlos de manera correcta e integrarlos de manera óptima.

En líneas generales se ha conseguido poder configurar y manejar de manera remota los dos instrumentos, se ha perfeccionado los conocimientos sobre LabVIEW y el manejo de esta herramienta, así como el conocimiento profundo de los instrumentos de un laboratorio.

6. Referencias

- [1] I. TORRES, I. TORRES and V. perfil, "ELEMENTOS DE UN SISTEMA DE MEDIDA", Ivansorzaingelectromecanica.blogspot.com, 2019. [Online]. Available: <http://ivansorzaingelectromecanica.blogspot.com/2014/09/elementos-de-un-sistema-de-medida.html>. [Accessed: 18- Feb- 2019].
- [2] "Mediciones eléctricas | Temario resuelto y completo", IngenieríaElectrónica, 2019. [Online]. Available: <https://ingenieriaelectronica.org/mediciones-electricas>. [Accessed: 18- Feb- 2019].
- [3] V. Garcia, "Circuito Diferenciador", Hispavila.com, 2019. [Online]. Available: <http://hispavila.com/total/3ds/tutores/circuitodiferenciador.htm> [Accessed: 18- Feb- 2019].
- [4] "History of GPIB - National Instruments", Ni.com, 2019. [Online]. Available: <http://www.ni.com/white-paper/3419/es/> [Accessed: 18- Feb- 2019].
- [5] "The Hewlett-Packard Interface Bus (HP-IB) GPIB IEEE-488 IEC625", Hp9845.net, 2019. [Online]. Available: <http://www.hp9845.net/9845/tutorials/hpib/> [Accessed: 18- Feb- 2019].
- [6] "GPIB IEEE 488 Connectors | Type 57 | Electronics Notes", Electronics-notes.com, 2019. [Online]. Available: <https://www.electronics-notes.com/articles/test-methods/gpib-ieee-488-bus/connectors.php> [Accessed: 18- Feb- 2019].
- [7] "PCI-GPIB Pinout - GPIB Instrument Control Device Manual - National Instruments", Ni.com, 2019. [Online]. Available: <http://www.ni.com/documentation/en/gpib-instrument-control-device/latest/pci-gpib/pinout/>. [Accessed: 18- Feb- 2019].
- [8] "SCPI Commands :: Radio-Electronics.Com", Radio-electronics.com, 2019. [Online]. Available: https://radio-electronics.com/info/t_and_m/scpi-standard-commands-for-programmable-instrumentation/commands-structure.php. [Accessed: 18- Feb- 2019].
- [9] "¿Que es un osciloscopio?", FinalTest.com, 2019. [Online]. Available: <https://www.finaltest.com.mx/product-p/art-9.htm>. [Accessed: 18- Feb- 2019].
- [10] Lázaro, A.M., "LabVIEW. Programación gráfica para el control de instrumentación", Paraninfo 2001.
- [11] *Model 2400 Series. Source Meter. User's Manual*. Keithley 2002.
- [12] *Yokogawa DL1520. User's Manual*. Yokogawa 1997.
- [13] "GPIB-USB-HS Pinout - GPIB Instrument Control Device Manual - National Instruments", Ni.com, 2019. [Online]. Available: <http://www.ni.com/documentation/en/gpib-instrument-control-device/latest/gpib-usb-hs/pinout/> [Accessed: 19- Feb- 2019].

[11]"GPIB-USB-HS+ - National Instruments", Ni.com, 2019. [Online]. Available: <http://www.ni.com/es-es/support/model.gpib-usb-hs-.html>. [Accessed: 18- Feb- 2019].

Anexo A: Presupuesto

Para el cálculo de este apartado vamos a tener en cuenta dos partes, la relativa al equipamiento y la relativa a las horas empleadas en la realización del proyecto.

- **Equipamiento necesario**

- Osciloscopio Yokogawa DL 1520: **353.21 €**
- SourceMeter Keithley 2410: **8890 €**
- Adaptador GPIB-USB-HS: **1102 €**
- Licencia LabVIEW 2017: **3451 €**

- **Horas de trabajo**

- Precio por hora del alumno: **25 €**
- Precio por hora del tutor: **40 €**

Ahora detallaremos todo en una tabla:

Concepto	Unidades	Horas	€/h	Precio
Equipamiento				
Osciloscopio Yokogawa DL1520	1	-	-	353,21 €
SourceMeter Keithley 2410	1	-	-	8.890 €
Adaptador GPIB-USB-HS	1	-	-	1.102 €
Licencia LabVIEW 2017	1	-	-	3.451 €
Descuento LabVIEW 2017 para estudiantes	1	-	-	-2.558 €
Personal				
Investigación y descarga de material necesario	-	40	25	1.000,00 €
Programación de aplicación	-	180	25	4.500,00 €
Realización memoria del proyecto	-	60	25	1.500,00 €
Tutoría personal con tutor	-	20	40	800,00 €
TOTAL PROYECTO				19.038,21 €

Tabla 1 Presupuesto

Como proyecto a ejecutar, el precio es bastante elevado teniendo en cuenta para lo que está enfocado. Cabe destacar que la fuente es el principal escollo de este proyecto, ya que con casi 9.000 € es casi la mitad del presupuesto. El proyecto económicamente hablando no es entable si se parte de cero. Si se disponen de los recursos materiales con anterioridad, si es viables ya que solo sería necesario abonar el trabajo del desarrollador.

Como alternativa de instrumentación, se puede elegir cualquier otra fuente que incluya comunicación GPIB. Se escogió la Keithley 2410 ya que era una de las más modernas que había en el laboratorio y que además contaba con GPIB.

Anexo B. Imágenes

Se incluyen a continuación algunas imágenes de funcionamiento.



Figura A.1 Fuente en funcionamiento

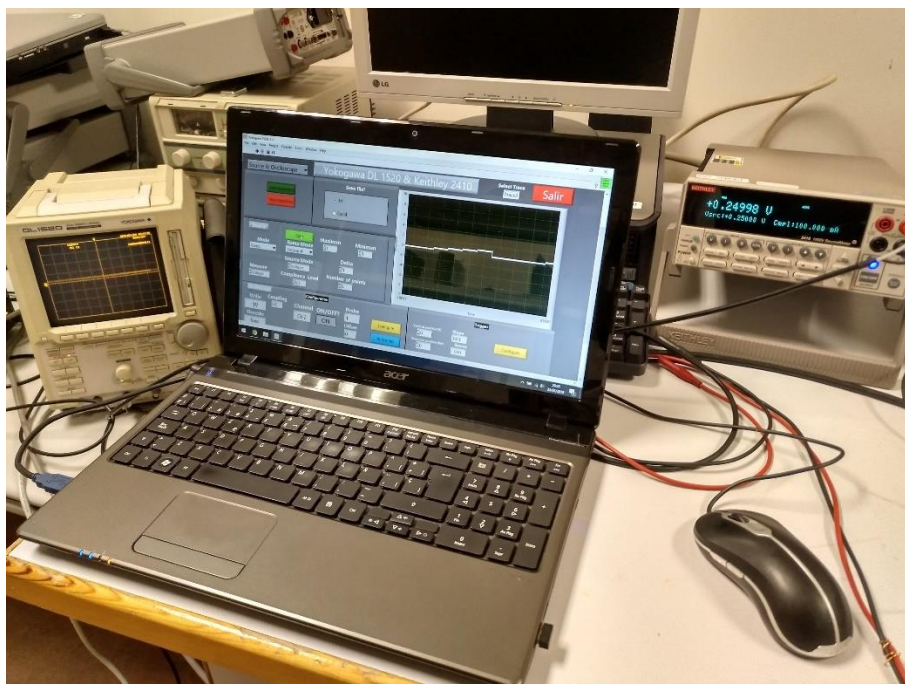


Figura A.2 Instrumentos y ordenador en marcha



Figura A.3 Detalle pantalla de osciloscopio acorde a figura 4.43



Figura A.4 Fuente externa para excitar señales cuadradas

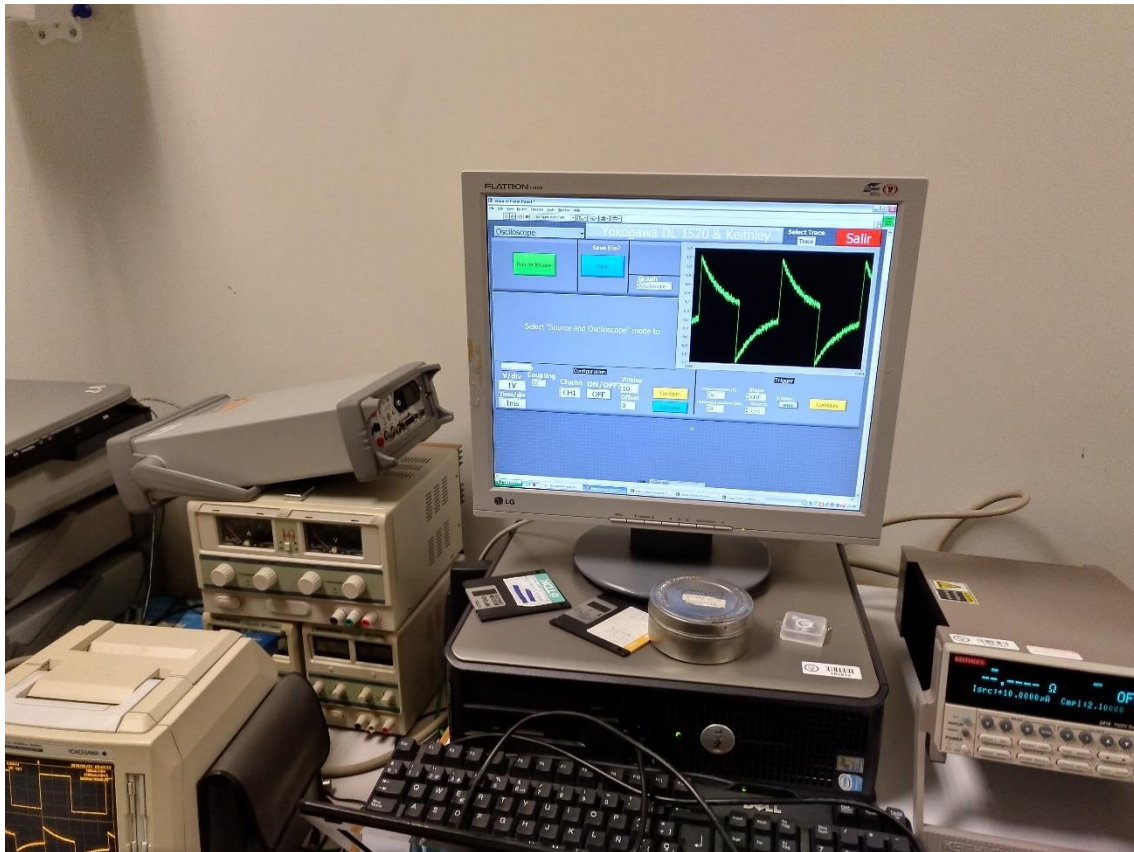


Figura A.5 Aplicación corriendo en otro terminal